



Программные интерфейсы

Руководство по эксплуатации

Содержание

| | |
|--|-----|
| CSTA API | 30 |
| Поддерживаемые методы/события..... | 30 |
| API по управлению конфигурацией ECSS-10 (http-terminal) | 41 |
| Сессионная работа HTTP-терминала | 41 |
| 1. Установление сессии..... | 41 |
| 2. Выполнение команд, периодические ping-запросы..... | 42 |
| 3. Завершение сессии | 42 |
| 4. Пример кода на Java для работы с HTTP-терминалом | 44 |
| Управление HTTP-терминалом..... | 46 |
| Подразделы | 46 |
| Hc_aaa_access_options - Управление подсистемой RADIUS авторизации..... | 46 |
| Hc_aaa_access_options_clean | 46 |
| Hc_aaa_access_options_info..... | 48 |
| Hc_aaa_access_options_set | 52 |
| Hc_aaa_general_options - Управление общими настройками службы RADIUS AAA.. | 56 |
| Hc_aaa_general_options_clean | 57 |
| Hc_aaa_general_options_info | 59 |
| Hc_aaa_general_options_set | 63 |
| Hc_aaa_options - Управление службой RADIUS AAA | 66 |
| Hc_aaa_options_clean | 66 |
| Hc_aaa_options_info..... | 69 |
| Hc_aaa_options_set | 72 |
| Hc_adaptation_context - Управление контекстами адаптации | 75 |
| Hc_adaptation_context_delete | 76 |
| Hc_adaptation_context_import | 78 |
| Hc_adaptation_context_list..... | 81 |
| Hc_adaptation_context_show..... | 83 |
| Hc_address_book_options - Управление настройками подключения к БД "Адресная книга" | 98 |
| Hc_address_book_options_clean | 99 |
| Hc_address_book_options_info | 101 |
| Hc_address_book_options_set | 104 |

| | |
|---|-----|
| Hc_agent - Управление агентами виртуального call-центра..... | 107 |
| Hc_agent_clean | 108 |
| Hc_agent_declare..... | 112 |
| Hc_agent_info..... | 118 |
| Hc_agent_list | 121 |
| Hc_agent_remove..... | 124 |
| Hc_agent_set | 126 |
| Hc_alarms - Управление системой мониторинга и вывода аварий | 130 |
| Hc_ap_speaker_off..... | 130 |
| Hc_ap_status | 131 |
| Hc_alarms_list..... | 133 |
| Hc_alarms_mask_list | 148 |
| Hc_node_rps_alarms..... | 151 |
| Hc_alarms_mask_add | 152 |
| Hc_alarms_mask_del | 156 |
| Hc_alarms_clear | 158 |
| Hc_alarms_delete | 160 |
| Hc_alarm_notifiers_options_clean | 165 |
| Hc_alarm_notifiers_options_info..... | 167 |
| Hc_alarm_notifiers_options_set | 171 |
| Hc_alarms_notifiers_send_test_msg..... | 174 |
| Hc_alias_profile - Управление профилями алиасов..... | 177 |
| Hc_alias_profile_list | 177 |
| Hc_alias_profile_declare | 180 |
| Hc_alias_profile_remove..... | 182 |
| Hc_alias_profile_info | 184 |
| Hc_alias_profile_set | 186 |
| Hc_alias_profile_clean | 188 |
| Hc_aliases - Управление списками алиасов | 190 |
| Общая XSD-схема | 190 |
| Hc_aliases_list | 194 |
| Hc_aliases_list2 | 195 |
| Hc_alias - Управление алиасами..... | 199 |
| Hc_alias_list | 200 |
| Hc_alias_list2 | 203 |

| | |
|--|-----|
| Hc_alias_where | 206 |
| Hc_alias_who | 208 |
| Hc_alias_info | 210 |
| Hc_alias_set | 216 |
| Hc_alias_declare | 219 |
| Hc_alias_remove..... | 222 |
| Hc_alias_clean | 224 |
| Hc_aon_category_options - Команды мапинга категории АОН в категорию ОКС | 227 |
| aon_category_options_clean..... | 227 |
| aon_category_options_info | 231 |
| aon_category_options_set..... | 236 |
| Hc_backup - Команды копирования и восстановления настроек | 239 |
| Hc_backup_restore_domain..... | 239 |
| Hc_backup_restore..... | 241 |
| Hc_backup_store_domain | 242 |
| Hc_backup_store | 244 |
| Hc_bridge - Управление бриджами | 246 |
| Hc_bridge_declare..... | 246 |
| Hc_bridge_info..... | 250 |
| Hc_bridge_remove | 252 |
| Hc_bridge_cleanup..... | 254 |
| Hc_bridge_change | 254 |
| Hc_calendar_timetable - Команды управления календарем и расписанием | 257 |
| Hc_calendar_timetable_show | 257 |
| Hc_calendar_timetable_set..... | 259 |
| Hc_calendar_timetable_remove | 261 |
| Hc_calls_active - Управление сервисом service actual calls | 263 |
| Hc_calls_active | 263 |
| Hc_calls_history - Управление историей вызовов..... | 267 |
| Hc_calls_history..... | 268 |
| Hc_purge_calls_db | 276 |
| Hc_cc_csta_server_options - Управление настройками подключения по протоколу CSTA..... | 279 |
| Hc_cc_csta_server_options_clean | 279 |

| | |
|--|-----|
| Hc_cc_csta_server_options_info | 281 |
| Hc_cc_csta_server_options_set | 285 |
| Hc_cc_queue_acw_status_set - Команды управления наборами статусов постобработки | 288 |
| Hc_cc_queue_acw_status_set_declare | 288 |
| Hc_cc_queue_acw_status_set_remove | 290 |
| Hc_cc_queue_acw_status_set_info | 292 |
| Hc_cc_queue_acw_status_add_status | 294 |
| Hc_cc_queue_acw_status_set_remove_status | 296 |
| Hc_cc_queue_acw_status - Команды управления статусами постобработки | 299 |
| Hc_cc_queue_acw_status_declare | 300 |
| Hc_cc_queue_acw_status_remove | 302 |
| Hc_cc_queue_acw_status_info | 304 |
| Hc_cc_queue_acw_status_add_label | 306 |
| Hc_cc_queue_acw_status_remove_label | 308 |
| Hc_cc_queue_agent_auxwork_reason - Команды управления причинами технического перерыва агента Call-центра | 311 |
| Hc_cc_queue_agent_auxwork_reason_add | 311 |
| Hc_cc_queue_agent_auxwork_reason_clean | 314 |
| Hc_cc_queue_agent_auxwork_reason_info | 316 |
| Hc_cc_queue_agent_auxwork_reason_remove | 320 |
| Hc_cc_queue_agent_auxwork_reason_set | 322 |
| Hc_cc_queue_agent_profile - Команды изменения причин технического перерыва для профиля агента call-центра | 324 |
| Hc_cc_queue_agent_profile_add | 325 |
| Hc_cc_queue_agent_profile_clean | 327 |
| Hc_cc_queue_agent_profile_info | 329 |
| Hc_cc_queue_agent_profile_remove | 333 |
| Hc_cc_queue_agent_profile_set | 335 |
| Hc_cfc_options - Управление настройками сервиса CFC | 337 |
| Hc_cfc_options_clean | 338 |
| Hc_cfc_options_info | 340 |
| Hc_cfc_options_set | 345 |
| Hc_change_context - Команда импорта контекста маршрутизации | 349 |
| Пример | 352 |

| | |
|--|-----|
| Hc_cluster_statistics - Управление системой сбора статистики для кластера..... | 357 |
| Hc_cluster_statistics_add | 357 |
| Hc_cluster_statistics_delete | 361 |
| Hc_cluster_statistics_statmodinfo | 362 |
| Hc_cluster_statistics_addcolmap..... | 365 |
| Hc_cluster_statistics_delcolmap..... | 367 |
| Hc_cluster_statistics_list..... | 368 |
| Hc_cluster - Управление кластерами | 370 |
| Hc_cluster_info | 370 |
| Hc_list_of_adapter_clusters | 374 |
| Hc_get_cluster_properties..... | 376 |
| Hc_cluster_set..... | 378 |
| Hc_cluster_clear | 380 |
| Hc_clusters_list..... | 382 |
| Hc_cluster_properties | 384 |
| Hc_cn_options - Управление службой нотификации вызова CN | 386 |
| Hc_cn_options_clean | 386 |
| Hc_cn_options_info..... | 389 |
| Hc_cn_options_set | 393 |
| Hc_cocon_configure - Управление CoCon-сессией..... | 396 |
| Hc_cocon_configure_info | 396 |
| Hc_cocon_configure_start | 398 |
| Hc_cocon_configure_stop | 400 |
| Hc_cocon - Управление пользователями системы | 402 |
| Hc_cocon_add_user | 403 |
| Hc_cocon_del_user | 407 |
| Hc_cocon_add_user_to_group..... | 409 |
| Hc_cocon_del_user_from_group | 412 |
| Hc_my_groups | 416 |
| Hc_cocon_passwd | 418 |
| Hc_cocon_check_passwd | 420 |
| Hc_cocon_users_list..... | 422 |
| Hc_cocon_groups_list | 426 |
| Hc_cocon_roles_list | 430 |
| Hc_cocon_user_property_clean..... | 432 |

| | |
|--|-----|
| Hc_cocon_user_property_info | 434 |
| Hc_cocon_user_property_set..... | 437 |
| Hc_conference_list - Команда просмотра участников конференции/конференц-комнаты..... | 439 |
| Hc_core_ai - Команды управления акустическими сигналами | 443 |
| Hc_core_ai_clean..... | 443 |
| Hc_core_ai_info | 445 |
| Hc_core_ai_set..... | 457 |
| Hc_core_messages - Управление настройками сообщений..... | 465 |
| Hc_core_messages_clean | 465 |
| Hc_core_messages_info..... | 467 |
| Hc_core_messages_set | 471 |
| Hc_core_timers - Управление таймерами виртуальной АТС | 474 |
| Hc_core_timers_clean..... | 474 |
| Hc_core_timers_info | 477 |
| Hc_core_timers_set..... | 481 |
| Hc_custom_cb - Управление виджетом сервиса "custom callback" | 484 |
| Hc_custom_cb_new | 484 |
| Hc_custom_cb_set | 486 |
| Hc_custom_cb_list | 488 |
| Hc_custom_cb_remove..... | 490 |
| Hc_dialer_campaign - Управление кампаниями дозвона | 492 |
| Hc_dialer_campaign_declare | 492 |
| Hc_dialer_campaign_info | 497 |
| Hc_dialer_campaign_action | 501 |
| Hc_dialer_campaign_list..... | 504 |
| Hc_dialer_campaign_start..... | 507 |
| Hc_dialer_db_cluster_options - Управление параметрами подключения в БД подсистемы автообзвонщика | 510 |
| Hc_dialer_db_cluster_options_clean..... | 511 |
| Hc_dialer_db_cluster_options_info | 513 |
| Hc_dialer_db_cluster_options_set..... | 517 |
| Hc_dialer_report_number - Команда получения отчета по определенному номеру в автообзвоне | 520 |
| Hc_dialer_report_number_info | 520 |

| | |
|---|-----|
| Hc_dialer_template - Управление шаблонами дозвона..... | 522 |
| Hc_dialer_template_declare..... | 523 |
| Hc_dialer_template_info..... | 527 |
| Hc_dialer_template_set..... | 530 |
| Hc_dialer_template_list..... | 533 |
| Hc_dialer_template_remove..... | 535 |
| Hc_direction - Управление направлениями | 538 |
| Hc_direction_list..... | 538 |
| Hc_direction_declare | 540 |
| Hc_direction_remove | 544 |
| Hc_direction_set..... | 546 |
| Hc_direction_info | 551 |
| Hc_direction_reset_stat..... | 554 |
| Hc_direction_clean..... | 557 |
| Hc_distinctive_ring_rule - Команды управления услугой distinctive_ring | 560 |
| Hc_distinctive_ring_rule_info..... | 560 |
| Hc_distinctive_ring_rule_remove..... | 562 |
| Hc_distinctive_ring_rule_set | 564 |
| Hc_domain_access_type - Команды управления типами доступа | 566 |
| Hc_domain_access_type_info | 566 |
| Hc_domain_access_type_remove..... | 568 |
| Hc_domain_access_type_declare | 569 |
| Hc_domain_calendar - Управление календарем на уровне домена..... | 571 |
| Hc_domain_calendar_set | 571 |
| Hc_domain_calendar_show | 576 |
| Hc_domain_call_statistics - Команда для получения значения общей статистики выззовов домена..... | 581 |
| Hc_domain_call_statistics | 581 |
| Hc_domain_cc_options - Управление настройками ограничений колл-центра домена | 585 |
| Hc_domain_cc_options_clean..... | 585 |
| Hc_domain_cc_options_info | 588 |
| Hc_domain_cc_options_set..... | 592 |
| Hc_domain_data - Управление свойствами домена | 596 |
| Hc_domain_data_info | 596 |

| | |
|--|-----|
| Hc_domain_data_set | 599 |
| Hc_domain_data_clean | 603 |
| Hc_domain_group - Управление группами доступа..... | 606 |
| Hc_domain_group_list Hc_domain_group_add Hc_domain_group_remove Hc_domain_group_add_access Hc_domain_group_remove_access Hc_domain_group_access_matrix Hc_domain_group_list..... | 606 |
| Hc_domain_group_add..... | 608 |
| Hc_domain_group_remove | 609 |
| Hc_domain_group_add_access..... | 610 |
| Hc_domain_group_remove_access | 611 |
| Hc_domain_group_access_matrix | 612 |
| Hc_domain_ldap - Команды управления LDAP/AD сервером | 614 |
| hc_domain_ldap_declare..... | 614 |
| hc_domain_ldap_info | 617 |
| hc_domain_ldap_remove | 619 |
| hc_domain_ldap_set..... | 621 |
| Hc_domain_limits - Управление ограничениями в домене | 624 |
| Hc_domain_limits_clean | 624 |
| Hc_domain_limits_info | 628 |
| Hc_domain_limits_set..... | 632 |
| Hc_domain_pbx_create - Создание PBX-домена с предопределенными настройками | 639 |
| Hc_domain_pbx_create | 639 |
| Hc_domain_properties - Управление свойствами виртуальной АТС..... | 644 |
| Hc_domain_properties_info | 644 |
| Hc_domain_properties_set..... | 651 |
| Hc_domain_properties_clean..... | 659 |
| Hc_domain_radius_server - Настройка RADIUS серверов на уровне домена | 663 |
| Hc_domain_radius_servers_info | 663 |
| Hc_domain_radius_servers_set | 666 |
| Hc_domain_regime - Управление режимами обслуживания абонентов | 669 |
| Hc_domain_import_regime..... | 669 |
| Hc_domain_regime_info..... | 671 |
| Hc_domain_regime_remove..... | 673 |
| Hc_domain_regime_declare..... | 674 |

| | |
|--|-----|
| Hc_domain_statistics - Управление системой сбора статистики для виртуальной АТС..... | 676 |
| Hc_domain_statistics_add..... | 676 |
| Hc_domain_statistics_delete..... | 678 |
| Hc_domain_statistics_statmodinfo | 679 |
| Hc_domain_statistics_addcolmap..... | 682 |
| Hc_domain_statistics_delcolmap | 684 |
| Hc_domain_statistics_list..... | 685 |
| Hc_domain - Управление виртуальными АТС..... | 687 |
| Hc_domain_declare..... | 688 |
| Hc_list_of_domains..... | 691 |
| Hc_domain_remove | 693 |
| Hc_domain_general_info..... | 694 |
| Hc_domain_copy..... | 698 |
| Hc_ds_cmds_list - Команды управления списками..... | 700 |
| Hc_ds_cmds_lists_declare..... | 701 |
| Hc_ds_cmds_lists_info..... | 703 |
| Hc_ds_cmds_lists_list..... | 706 |
| Hc_ds_cmds_lists_number_add | 708 |
| Hc_ds_cmds_lists_number_remove..... | 711 |
| Hc_ds_cmds_lists_remove | 713 |
| Hc_ds_cmds_lists_set..... | 715 |
| Hc_ds_domain_check_addr_to_digitmap_cmd - Проверка адреса по DigitMap'у..... | 717 |
| Hc_ds_general_options - Управление свойствами DS (системные свойства)..... | 719 |
| Общая схема..... | 719 |
| Hc_ds_general_options_info | 720 |
| Hc_ds_general_options_set..... | 722 |
| Hc_ds_general_options_clean..... | 724 |
| Hc_ecss_control_channel_options - Управление подсистемой RADIUS авторизации..... | 726 |
| Hc_ecss_control_channel_options_clean | 726 |
| Hc_ecss_control_channel_options_info..... | 729 |
| Hc_ecss_control_channel_options_set | 732 |
| Hc_ecss_mlpp_options - Управление многоадресной приоритизацией вызовов .. | 736 |
| Hc_ecss_mlpp_options_clean | 736 |

| | |
|---|-----|
| Hc_ecss_mlpp_options_info..... | 738 |
| Hc_ecss_mlpp_options_set..... | 743 |
| Hc_ecss_statistics - Получение статистики по домену..... | 747 |
| Hc_ecss_statistics_show..... | 747 |
| Hc_ecss_statistics_get_metric..... | 753 |
| Hc_ftp_cluster_options - Управление настройками FTP подсистемы TTS..... | 756 |
| Hc_ftp_cluster_options_info..... | 756 |
| Hc_ftp_cluster_options_set..... | 761 |
| Hc_ftp_cluster_options_clean..... | 764 |
| Hc_gateway_port - Управление портами шлюза..... | 766 |
| Hc_gateway_port_info..... | 767 |
| Hc_gateway_port_set..... | 770 |
| Hc_gateway_port_clean..... | 771 |
| Hc_gateway_profile - Управление профилями настроек шлюза..... | 773 |
| Hc_gateway_profile_declare..... | 773 |
| Hc_gateway_profile_info..... | 774 |
| Hc_gateway_profile_set..... | 777 |
| Hc_gateway_profile_clean..... | 778 |
| Hc_gateway_profile_remove..... | 779 |
| Hc_gateway_sdp - Управление SDP-шаблонами..... | 781 |
| Hc_gateway_sdp_template_declare..... | 781 |
| Hc_gateway_sdp_template_remove..... | 783 |
| Hc_gateway_sdp_template_info..... | 784 |
| Hc_gateway_sdp_template_set..... | 786 |
| Hc_gateway - Управление шлюзами..... | 789 |
| Hc_gateway_declare..... | 789 |
| Hc_gateway_info..... | 791 |
| Hc_gateway_remove..... | 795 |
| Hc_gateway_set..... | 796 |
| Hc_gateway_clean..... | 797 |
| Hc_gateway_add_port..... | 798 |
| Hc_gateway_remove_port..... | 799 |
| Hc_gateway_add_sdp_template..... | 800 |
| Hc_gateway_remove_sdp_template..... | 802 |
| Hc_gateway_change_profile..... | 803 |

| | |
|---|-----|
| Hc_get_extended_subscriber_card - Вывод информационной карточки абонента. | 804 |
| Hc_get_extended_subscriber_card..... | 804 |
| Hc_get_info - Время работы узла(ов) с момента последнего перезапуска | 811 |
| Hc_get_info..... | 811 |
| Hc_hosts_info - Информация о состоянии серверов..... | 814 |
| hosts_info..... | 814 |
| Hc_http_routing_server - Команды управления HTTP серверами маршрутизации..... | 818 |
| Hc_http_routing_servers_change | 818 |
| Hc_http_routing_servers_declare | 821 |
| Hc_http_routing_servers_info..... | 823 |
| Hc_http_routing_servers_remove..... | 826 |
| Hc_http_routing_servers_set_all | 828 |
| Hc_iface - Управление интерфейсами | 830 |
| Hc_iface_group_owner_info..... | 830 |
| Hc_iface_group_owner_list | 835 |
| Hc_iface_list..... | 838 |
| Hc_iface_info | 843 |
| Hc_iface_user_set..... | 847 |
| Hc_iface_safe_user_set..... | 851 |
| Hc_isup_cause_messages - управление текстовыми описаниями для ISUP cause .. | 853 |
| Hc_isup_cause_messages_clean..... | 853 |
| Hc_isup_cause_messages_info | 855 |
| Hc_isup_cause_messages_set..... | 862 |
| Hc_ivr_blocks_outputs_get - Команда просмотра блоков IVR-скрипта | 867 |
| Hc_ivr_blocks_outputs_get | 867 |
| Hc_ivr_script_restriction - Управление ограничениями IVR скриптов | 869 |
| Hc_ivr_scripts_restriction_profile_info..... | 869 |
| Hc_ivr_scripts_restriction_profile_apply..... | 873 |
| Hc_ivr_scripts_restriction_profile_set | 874 |
| Hc_ivr_scripts_restriction_profile_remove | 878 |
| Hc_ivr_scripts_restriction_options_info..... | 880 |
| Hc_ivr_scripts_restriction_options_set | 883 |
| Hc_ivr_script - Управление IVR скриптами..... | 886 |
| Hc_ivr_script_delete..... | 886 |

| | |
|--|-----|
| Hc_ivr_script_import..... | 889 |
| Hc_ivr_script_list..... | 893 |
| Hc_ivr_script_show..... | 896 |
| Hc_ivr_script_validate..... | 899 |
| Hc_ivr_script_rename..... | 902 |
| Hc_ivr_variables - Команды просмотра списка переменных IVR-скрипта..... | 904 |
| Hc_ivr_undeclared_variables_get..... | 904 |
| Hc_ivr_declared_variables_get..... | 906 |
| Hc_licence - Управление лицензиями (HTTP-терминал)..... | 908 |
| Hc_add_licence..... | 908 |
| Hc_current_limits..... | 911 |
| Hc_list_licence..... | 915 |
| Hc_delete_licence..... | 924 |
| Hc_set_passport..... | 926 |
| Hc_show_passport..... | 929 |
| Hc_md_rpss - Управление свойствами кластера Mediator (RPSS)..... | 932 |
| Hc_md_rpss_info..... | 932 |
| Hc_md_rpss_set..... | 936 |
| Hc_md_rpss_clean..... | 940 |
| Hc_media_profile - Управление медиа-профилями на виртуальной АТС..... | 942 |
| Hc_media_profile_list..... | 943 |
| Hc_media_profile_declare..... | 946 |
| Hc_media_profile_info..... | 950 |
| Hc_media_profile_set..... | 952 |
| Hc_media_profile_remove..... | 955 |
| Hc_media_registrar - Команды управления регистратором для MSR медиа менеджера..... | 957 |
| Hc_media_registrar_clean..... | 958 |
| Hc_media_registrar_info..... | 960 |
| Hc_media_registrar_set..... | 962 |
| Hc_media_resource - Команды управления медиа-ресурсами..... | 966 |
| Hc_media_resource_declare..... | 966 |
| Hc_media_resource_info..... | 969 |
| Hc_media_resource_purge..... | 973 |
| Hc_media_resource_remove..... | 975 |

| | |
|---|------|
| Hc_media_resource_set..... | 978 |
| Hc_media_site - Команды управления географическими зонами обслуживания медиа-сервера | 982 |
| Hc_media_site_declare | 982 |
| Hc_media_site_distance | 986 |
| Hc_media_site_list | 989 |
| Hc_media_site_matrix..... | 991 |
| Hc_media_site_remove..... | 993 |
| Hc_media_zone - Команды управления идентификатором сети | 996 |
| Hc_media_zone_declare | 997 |
| Hc_media_zone_list | 1000 |
| Hc_media_zone_remove..... | 1002 |
| Hc_mediator_http_terminal - Управление подсистемой HTTP Terminal..... | 1004 |
| Общая схема | 1005 |
| Hc_mediator_http_terminal_info..... | 1005 |
| Hc_mediator_http_terminal_set | 1008 |
| Hc_mediator_http_terminal_clean | 1011 |
| Hc_megaco_gateways - Управление настройками Megaco-шлюзов..... | 1014 |
| Hc_megaco_gateways_add | 1014 |
| Hc_megaco_gateways_list..... | 1016 |
| Hc_megaco_gateways_remove | 1018 |
| Hc_megaco - Управление настройками RA Megaco | 1019 |
| Hc_megaco_info | 1019 |
| Hc_megaco_set..... | 1023 |
| Hc_megaco_clean..... | 1027 |
| Hc_modifiers_context - Управление контекстами модификации..... | 1030 |
| Hc_modifiers_context_delete | 1030 |
| Hc_modifiers_context_import..... | 1032 |
| Hc_modifiers_context_list..... | 1035 |
| Hc_modifiers_context_show | 1037 |
| Hc_msr_timers_info - Информация о таймерах MSR..... | 1057 |
| Hc_msr_timers_info | 1057 |
| hc_node_last_errors - Список N последних error-логов..... | 1060 |
| Hc_node_rps - Управление подсистемой сбора статистики и предупреждений ноды..... | 1064 |

| | |
|---|-------------|
| Hc_node_rps_alarms..... | 1065 |
| Hc_node_rps_controls | 1066 |
| Hc_node_rps_events | 1067 |
| Hc_node_rps_logs | 1068 |
| Hc_node_rps_stats | 1068 |
| Hc_node_rps_status..... | 1069 |
| Hc_node - Управление нодами | 1071 |
| Hc_node_memory_info..... | 1071 |
| Hc_node_application_info..... | 1073 |
| Hc_node_full_applications_info..... | 1075 |
| Hc_node_releases_info | 1083 |
| Hc_node_statistics_info | 1085 |
| Hc_node_system_info | 1088 |
| Hc_node_service..... | 1098 |
| Hc_node_show_cert..... | 1099 |
| Hc_notifier_options - Управление подсистемой нотификации..... | 1101 |
| Hc_notifier_options_clean | 1102 |
| Hc_notifier_options_info | 1104 |
| Hc_notifier_options_set..... | 1109 |
| Hc_notifier_send_test - Управление отправкой нотификаций через службу уведомлений..... | 1113 |
| Hc_notifier_send_test_msg | 1113 |
| Hc_np_numbers - Управление списками номеров плана нумерации..... | 1115 |
| Hc_np_add_number..... | 1115 |
| Hc_np_delete_number2..... | 1119 |
| Hc_np_rename_number | 1123 |
| Hc_np_list_number | 1126 |
| Hc_np_bind_number..... | 1129 |
| Hc_np_unbind_number | 1134 |
| Hc_np - Управление настройками плана нумерации | 1139 |
| Hc_np_declare | 1139 |
| Hc_np_info | 1142 |
| Hc_np_set..... | 1144 |
| Hc_np_clean..... | 1147 |
| Hc_np_remove..... | 1151 |

| | |
|--|------|
| Hc_np_list..... | 1153 |
| Hc_np_db_list..... | 1155 |
| Hc_oct_cp_options - Управление системными настройками ecss-core..... | 1157 |
| Hc_oct_cp_options_clean..... | 1157 |
| Hc_oct_cp_options_info | 1159 |
| Hc_oct_cp_options_set..... | 1162 |
| Hc_queue - Управление очередями виртуального call-центра..... | 1165 |
| Hc_queue_add_agent | 1166 |
| Hc_queue_clean | 1169 |
| Hc_queue_configuration..... | 1171 |
| Hc_queue_declare..... | 1176 |
| Hc_queue_list | 1182 |
| Hc_queue_remove..... | 1185 |
| Hc_queue_remove_agent..... | 1187 |
| Hc_queue_set | 1190 |
| Hc_restfs_clusters_autocleaner - Управление настройками подсистемы Autocleaner кластера RestFS | 1193 |
| Hc_restfs_clusters_autocleaner_add_rule | 1193 |
| Hc_restfs_clusters_autocleaner_change_rule..... | 1195 |
| Hc_restfs_clusters_autocleaner_info..... | 1197 |
| Hc_restfs_clusters_autocleaner_remove_rule..... | 1200 |
| Hc_restfs_clusters - Управление RestFS кластерами на уровне ECSS-10 | 1202 |
| Hc_restfs_clusters_declare | 1202 |
| Hc_restfs_clusters_list..... | 1204 |
| Hc_restfs_clusters_remove..... | 1206 |
| Hc_restfs_clusters_status | 1208 |
| Hc_restfs_clusters_set..... | 1210 |
| Hc_restfs_domain_autocleaner - Управление настройками подсистемы Autocleaner кластера RestFS на уровне домена..... | 1212 |
| Hc_restfs_domain_autocleaner_add_rule..... | 1212 |
| Hc_restfs_domain_autocleaner_change_rule..... | 1214 |
| Hc_restfs_domain_autocleaner_remove_rule | 1217 |
| Hc_restfs_domain_autocleaner_info | 1220 |
| Hc_restfs_domain_autocleaner_set..... | 1222 |
| Hc_restfs_domain_list - Просмотр списка файлов по указанному пути на определенном домене | 1225 |

| | |
|---|------|
| Hc_restfs_domain_list..... | 1225 |
| Hc_restfs_list_clusters - Просмотр списка файлов на кластере Restfs..... | 1227 |
| Hc_restfs_list_clusters..... | 1227 |
| Hc_restfs_list - Просмотр списка файлов на RestFS для системы или виртуальной АТС..... | 1234 |
| Hc_restfs_list..... | 1234 |
| Hc_restfs_url..... | 1244 |
| Hc_role - Команды управления ролями..... | 1247 |
| Hc_role_add_group..... | 1247 |
| Hc_role_declare..... | 1249 |
| Hc_role_del_group..... | 1251 |
| Hc_role_properties_clean..... | 1253 |
| Hc_role_properties_info..... | 1256 |
| Hc_role_properties_set..... | 1262 |
| Hc_role_remove..... | 1266 |
| Hc_set_role_to_user..... | 1268 |
| Hc_del_role_from_user..... | 1270 |
| Hc_routing - Управление маршрутизацией..... | 1272 |
| Hc_change_context..... | 1272 |
| Hc_list_of_context..... | 1275 |
| Hc_show_context..... | 1278 |
| Hc_delete_context..... | 1297 |
| Hc_security_profile - команды управления профилями безопасности..... | 1298 |
| Hc_security_profile_clean..... | 1298 |
| Hc_security_profile_info..... | 1300 |
| Hc_security_profile_set..... | 1303 |
| Hc_sip_cluster_network - Управление сетевыми настройками SIP адаптера в кластере..... | 1306 |
| Hc_sip_cluster_network_clean..... | 1306 |
| Hc_sip_cluster_network_info..... | 1308 |
| Hc_sip_cluster_network_set..... | 1311 |
| Hc_sip_domain_network - Управление сетевыми настройками протокола SIP на уровне домена..... | 1314 |
| Hc_sip_domain_network_info..... | 1314 |
| Hc_sip_domain_network_clean..... | 1317 |

| | |
|---|------|
| Hc_sip_domain_network_set | 1319 |
| Hc_sip_domain_properties - Управление свойствами SIP на уровне виртуальной АТС | 1323 |
| Hc_sip_domain_properties_clean | 1323 |
| Hc_sip_domain_properties_info | 1325 |
| Hc_sip_domain_properties_set | 1330 |
| Hc_sip_dynamic_user_area - Команды управления динамически конфигурируемыми интерфейсами | 1334 |
| Hc_sip_dynamic_user_area_info | 1334 |
| Hc_sip_dynamic_user_area_set | 1337 |
| Hc_sip_error_messages - управление текстовыми описаниями ошибок SIP | 1340 |
| Hc_sip_error_messages_clean | 1340 |
| Hc_sip_error_messages_info | 1342 |
| Hc_sip_error_messages_set | 1347 |
| Hc_sip_general - Управление общими свойствами кластера SIP | 1350 |
| Hc_sip_general_info | 1350 |
| Hc_sip_general_set | 1354 |
| Hc_sip_general_clean | 1357 |
| Hc_sip_internal_messages - Управление текстовыми описаниями для внутренних кодов SIP | 1359 |
| sip_internal_messages_clean | 1360 |
| sip_internal_messages_info | 1363 |
| sip_internal_messages_set | 1369 |
| Hc_sip_ldap - Управление параметрами соединения с сервером LDAP | 1372 |
| Hc_sip_ldap_info | 1372 |
| Hc_sip_ldap_set | 1376 |
| Hc_sip_ldap_clean | 1380 |
| Hc_sip_number_modifications - Формирование правил модификации номера ... | 1383 |
| Hc_sip_number_modifications_clean | 1383 |
| Hc_sip_number_modifications_info | 1385 |
| Hc_sip_number_modifications_set | 1388 |
| Hc_sip_pcap_trace - Управление трассировкой | 1392 |
| Hc_sip_pcap_trace_start | 1392 |
| Hc_sip_pcap_trace_stop | 1394 |
| Hc_sip_status_messages - Управление текстовыми описаниями для SIP-cause ... | 1396 |

| | |
|--|------|
| Hc_sip_status_messages_clean | 1396 |
| Hc_sip_status_messages_info | 1398 |
| Hc_sip_status_messages_set | 1404 |
| Hc_sip_telephone_reboot - Перезагрузка телефонных аппаратов..... | 1407 |
| Hc_sip_telephone_reboot..... | 1408 |
| Hc_sip_timers - Управление таймерами кластера SIP | 1410 |
| Hc_sip_timers_info..... | 1410 |
| Hc_sip_timers_set | 1413 |
| Hc_sip_timers_clean | 1416 |
| Hc_sip_transport - Управление транспортными свойствами кластера SIP | 1419 |
| Hc_sip_transport_info..... | 1419 |
| Hc_sip_transport_set | 1421 |
| Hc_sip_transport_clean | 1425 |
| Hc_sip_trunk - Управление SIP-транками (HTTP-терминал) | 1427 |
| Hc_sip_trunk_declare..... | 1427 |
| Hc_sip_trunk_set | 1430 |
| Hc_sip_trunk_show | 1435 |
| Hc_sip_trunk_remove | 1439 |
| Hc_sip_trunk_check | 1440 |
| Hc_sip_user - Управление SIP-абонентами..... | 1443 |
| Hc_sip_user_declare | 1443 |
| Hc_sip_user_show | 1448 |
| Hc_sip_user_registered..... | 1453 |
| Hc_sip_user_set..... | 1458 |
| Hc_sip_user_remove | 1462 |
| Hc_sip_user_check..... | 1465 |
| Hc_snmp_agent_system_options - Управление настройками SNMP на уровне медиатор-а | 1467 |
| Hc_snmp_agent_system_options_clean | 1468 |
| Hc_snmp_agent_system_options_info..... | 1470 |
| Hc_snmp_agent_system_options_set | 1473 |
| Hc_sorm_options - Управление доменными настройками СОРМ | 1476 |
| Hc_sorm_options_clean..... | 1477 |
| Hc_sorm_options_info | 1480 |
| Hc_sorm_options_set..... | 1483 |

| | |
|--|------|
| Hc_sorm_system_options - Управление системными свойствами СОРМ | 1487 |
| Hc_sorm_system_options_clean..... | 1488 |
| Hc_sorm_system_options_info | 1491 |
| Hc_sorm_system_options_set..... | 1494 |
| Hc_sp_api -Команды httpTerminal-а для работы с сессией пользователя в Портале абонента | 1497 |
| Hc_sp_api_login | 1497 |
| Hc_sp_api_logout..... | 1499 |
| Hc_sp_api_change_login | 1501 |
| Hc_sp_api_change_password | 1503 |
| Hc_sp_api_check_login..... | 1505 |
| Hc_sp_api_check_session..... | 1507 |
| Hc_sp - Управление порталом (кабинетом) абонента..... | 1509 |
| Hc_check_connection | 1509 |
| Hc_sp_properties_list..... | 1512 |
| Hc_sp_properties_set..... | 1515 |
| Hc_sp_properties_clean..... | 1518 |
| Hc_sync | 1521 |
| Hc_ss_alias_package_list - Команда для просмотра пакетов ДВО абонента..... | 1524 |
| Hc_ss_package_list | 1524 |
| Hc_ss_domain_profile - Управление профилями услуг | 1527 |
| Hc_ss_domain_profile_activate | 1527 |
| Hs_ss_domain_profile_add_ss..... | 1530 |
| Hc_ss_domain_profile_copy..... | 1532 |
| Hc_ss_domain_profile_info | 1535 |
| Hc_ss_domain_profile_list..... | 1569 |
| Hc_ss_domain_profile_remove..... | 1571 |
| Hc_ss_domain_profile_set..... | 1573 |
| Hc_ss_feature_code - Команды управления Feature кодами для работы с услугами..... | 1577 |
| Hc_ss_feature_code_clean | 1577 |
| Hc_ss_feature_code_info..... | 1581 |
| Hc_ss_feature_code_set | 1597 |
| Hc_ss_feature_code_properties_info | 1601 |
| Hc_ss_incall_feature_code - Управление списками feature кодов | 1603 |

| | |
|--|------|
| Hc_ss_incall_feature_code_clean | 1603 |
| Hc_ss_incall_feature_code_info..... | 1605 |
| Hc_ss_incall_feature_code_set | 1608 |
| Hc_ss_license_packages - Команды управления пакетами лицензий | 1612 |
| Hc_ss_licence_package_allocate | 1612 |
| Hc_ss_licence_package_free | 1614 |
| Hc_ss_licence_package_subscribers_list | 1617 |
| Hc_ss_package_limit - Команда для просмотра использования лицензий ДВО . | 1619 |
| Hc_ss_package_limit..... | 1619 |
| Hc_ss_preset_packages_info - Команда, возвращающая список ДВО с указанием их уровня..... | 1623 |
| Hc_ss_preset_packages_info..... | 1623 |
| Hc_ss_system_profile - Команды управления системным профилем | 1625 |
| Hc_ss_system_profile_copy | 1625 |
| Hc_ss_system_profile_copy_to_domain | 1629 |
| Hc_ss_system_profile_info | 1632 |
| Hc_ss_system_profile_list..... | 1666 |
| Hc_ss_system_profile_remove..... | 1668 |
| Hc_ss_system_profile_set..... | 1671 |
| Hc_ss_system_profile_add_ss..... | 1674 |
| Hc_ss_trunk - Команды управления транковыми сервисами..... | 1677 |
| Hc_ss_trunk_activate | 1677 |
| Hc_ss_trunk_deactivate..... | 1681 |
| Hc_ss_trunk_info | 1684 |
| Hc_ss_voicemail - Команды управления сервисом голосовой почты | 1700 |
| Hc_ss_voicemail_mailbox_modify..... | 1701 |
| Hc_ss_voicemail_mailbox_show..... | 1705 |
| Hc_ss - Управление услугами | 1707 |
| Общая схема управления услугами | 1708 |
| Hc_ss_acl_add | 1713 |
| Hc_ss_acl_remove | 1716 |
| Hc_ss_acl_show..... | 1717 |
| Hc_ss_activate | 1721 |
| Hc_ss_available_show..... | 1725 |
| Hc_ss_calls_history | 1760 |

| | |
|---|------|
| Hc_ss_deactivate | 1763 |
| Hc_ss_enable | 1765 |
| Hc_ss_enabled | 1769 |
| Hc_ss_info | 1772 |
| Hc_ss_install | 1809 |
| Hc_ss_show..... | 1810 |
| Hc_ss_uninstall..... | 1844 |
| Hc_statistics_show - Управление статистикой ECSS-10..... | 1846 |
| Hc_statistics_show | 1846 |
| Hc_sudo - Привилегированный режим | 1851 |
| Hc_sudo_cmd | 1852 |
| Hc_unsudo_cmd | 1853 |
| Hc_sys_iface_options - Управление настройками контекста маршрутизации для системных интерфейсов | 1854 |
| Hc_sys_iface_options_clean..... | 1854 |
| Hc_sys_iface_options_info | 1857 |
| Hc_sys_iface_options_set..... | 1860 |
| Hc_system_geo_options - Управление настройками ГЕО-резерва | 1863 |
| Hc_system_geo_options_clean | 1864 |
| Hc_system_geo_options_info..... | 1867 |
| Hc_system_geo_options_set | 1869 |
| Hc_tc_meetings - Управление историей совещаний..... | 1873 |
| Hc_tc_meetings_active_info..... | 1873 |
| Hc_tc_meetings_active_list | 1876 |
| Hc_tc_meetings_history..... | 1878 |
| Hc_tc_meetings_info..... | 1882 |
| Hc_tc_meetings_list | 1886 |
| Hc_tc_meeting - Управление шаблонами совещаний Teleconference на виртуальной АТС..... | 1888 |
| Hc_tc_meeting..... | 1888 |
| Hc_tc_meeting_remove | 1893 |
| Hc_tc_members_list - Просмотр списков всех участников и ведущих в домене. | 1895 |
| tc_members_list | 1895 |
| Hc_tc_options - Управление настройками телеконференции уровня кластера ядра..... | 1897 |

| | |
|--|------|
| Hc_tc_options_clean | 1897 |
| Hc_tc_options_info | 1901 |
| Hc_tc_options_set | 1904 |
| Hc_tc_phones - Управление телефонами сервиса "Селекторной связи" | 1907 |
| tc_phone_push | 1907 |
| tc_phone_status | 1909 |
| Hc_tc_system_options - Управление настройками Телеконференции на уровне системы..... | 1912 |
| Hc_tc_system_options_clean | 1912 |
| Hc_tc_system_options_info..... | 1916 |
| Hc_tc_system_options_set | 1918 |
| Hc_tc_web_options - Управление веб-свойствами телеконференции | 1922 |
| tc_web_options_clean | 1922 |
| tc_web_options_info..... | 1924 |
| tc_web_options_set | 1927 |
| Hc_tracer - Управление Tracer | 1931 |
| Hc_tracer_declare..... | 1931 |
| Hc_tracer_show | 1933 |
| Hc_tracer_remove | 1938 |
| Hc_trunk - Команды управления транками | 1939 |
| Hc_trunk_clean..... | 1939 |
| Hc_trunk_info | 1941 |
| Hc_trunk_set..... | 1944 |
| Hc_trunk_list..... | 1946 |
| Hc_tts_cluster_options - Управление опцией подсистемы TTS уровня кластера. | 1949 |
| Hc_tts_cluster_options_info..... | 1950 |
| Hc_tts_cluster_options_set | 1953 |
| Hc_tts_cluster_options_clean | 1956 |
| Hc_tts_domain_options - Управление опцией подсистемы TTS уровня домена .. | 1958 |
| Hc_tts_domain_options_info | 1958 |
| Hc_tts_domain_options_set..... | 1967 |
| Hc_tts_domain_options_clean..... | 1975 |
| Hc_tts_ftp_domain_options - Управления настройками FTP-пользователя..... | 1977 |
| Hc_tts_ftp_domain_options_info..... | 1977 |
| Hc_tts_ftp_domain_options_set | 1981 |

| | |
|--|------|
| Hc_tts_ftp_domain_optios_clean | 1984 |
| Hc_tts - Управление CDR..... | 1986 |
| Hc_tts_prefix | 1987 |
| Hc_tts_mode | 1988 |
| Hc_tts_add_cdr_group..... | 1990 |
| Hc_tts_change_cdr_group | 1992 |
| Hc_tts_show_cdr_group | 1996 |
| Hc_tts_info | 1999 |
| Hc_tts_remove_cdr_group | 2002 |
| Hc_rename_cdr_group..... | 2003 |
| Hc_tts_validate_cdr | 2005 |
| Hc_user_agent_list - команда просмотра списков user_agents | 2030 |
| Hc_valias - Управление виртуальными алиасами в рамках виртуальной АТС .. | 2033 |
| Hc_valias_declare | 2034 |
| Hc_valias_list | 2037 |
| Hc_valias_remove..... | 2039 |
| Hc_yandex_tts - Управление настройками Yandex TTS..... | 2041 |
| Hc_yandex_tts_clean | 2041 |
| Hc_yandex_tts_info | 2043 |
| Hc_yandex_tts_set | 2046 |
| API по работе с Call-центром | 2051 |
| Подразделы | 2051 |
| Метрики в режиме реального времени..... | 2052 |
| XSD файлы | 2053 |
| HTTP API | 2053 |
| Подразделы | 2053 |
| Команды HTTP API | 2053 |
| accept_fax.xsd..... | 2054 |
| add_participant.xsd..... | 2055 |
| agent_list.xsd | 2058 |
| all_queues.xsd | 2062 |
| auxwork.xsd | 2064 |
| block_operator.xsd..... | 2067 |
| call_consult.xsd | 2070 |

| | |
|--|------|
| call_history.xsd | 2072 |
| call_hold.xsd | 2075 |
| call_transfer.xsd..... | 2078 |
| call_unattended_transfer.xsd | 2079 |
| call_unhold.xsd | 2082 |
| cancel_force_logout.xsd | 2085 |
| cancel_procedure.xsd | 2086 |
| create_conference.xsd | 2088 |
| domain_list.xsd..... | 2092 |
| force_distribute.xsd | 2093 |
| force_logout.xsd | 2095 |
| get_commands_list.xsd..... | 2096 |
| group_list.xsd..... | 2096 |
| integration_list.xsd | 2099 |
| journal_list.xsd..... | 2103 |
| leave_conference.xsd | 2105 |
| make_agent_call.xsd | 2108 |
| make_available.xsd | 2111 |
| monitor.xsd | 2114 |
| move_conversation_to_queue.xsd | 2117 |
| operator_call_history.xsd | 2119 |
| ping.xsd | 2129 |
| queue_list.xsd | 2130 |
| read_user_data.xsd | 2132 |
| reject_consultation.xsd | 2136 |
| reject_fax.xsd | 2137 |
| reject_supervising.xsd | 2138 |
| remove_conference.xsd | 2139 |
| remove_conversation_from_queue.xsd | 2142 |
| remove_participant.xsd | 2144 |
| remove_user_data.xsd | 2147 |
| save_user_data.xsd | 2149 |
| send_to_email.xsd..... | 2152 |
| statistics_show.xsd | 2154 |
| supervise.xsd..... | 2163 |

| | |
|---|------|
| unlock_operator.xsd | 2165 |
| События HTTP API..... | 2168 |
| agent_info_event.xsd | 2168 |
| agent_list_event.xsd..... | 2169 |
| agent_realtime_statistics_event2.xsd..... | 2171 |
| agent_realtime_statistics_event.xsd..... | 2173 |
| auxwork_profile_event.xsd | 2176 |
| conference_participants_event2.xsd | 2180 |
| conference_participants_event.xsd | 2182 |
| conferences_event.xsd | 2184 |
| conversation.xsd | 2185 |
| conversations_event.xsd..... | 2189 |
| group_list_event.xsd | 2192 |
| group_realtime_statistics_event.xsd | 2194 |
| queue_config_event.xsd | 2196 |
| queue_conversations_event2.xsd | 2199 |
| queue_conversations_event.xsd | 2201 |
| queue_list_event.xsd..... | 2203 |
| queue_realtime_statistics_event.xsd..... | 2205 |
| supervising_event.xsd | 2206 |
| Вспомогательные схемы HTTP API | 2207 |
| agent.xsd: | 2208 |
| call_history_common.xsd: | 2210 |
| conversation.xsd:..... | 2212 |
| error.xsd: | 2214 |
| ss.xsd: | 2217 |
| Pubsub API | 2218 |
| Команды Pubsub API..... | 2218 |
| pubsub_login.xsd:..... | 2219 |
| События Pubsub API | 2219 |
| line_status_event.xsd:..... | 2219 |
| Подключение к ecss-cc-ui-api | 2220 |
| Использование технологии websocket..... | 2222 |
| Протокол общения | 2222 |
| Входящие запросы..... | 2223 |

| | |
|--|------|
| Ответы на запросы | 2223 |
| ecss-cc-ui API..... | 2225 |
| Команда auth..... | 2225 |
| Параметры | 2225 |
| Команда call/accept..... | 2226 |
| Параметры запроса | 2227 |
| Параметры ответа..... | 2227 |
| Команда call/complete | 2228 |
| Параметры запроса | 2228 |
| Параметры ответа..... | 2228 |
| Команда call/getCallRecord | 2229 |
| Параметры запроса | 2229 |
| Параметры ответа..... | 2230 |
| Команда call/makeCall | 2231 |
| Параметры запроса | 2231 |
| Параметры ответа..... | 2231 |
| Вызов по идентификатору оператора..... | 2232 |
| Команда call/reject | 2233 |
| Параметры запроса | 2233 |
| Параметры ответа..... | 2233 |
| Команда call/subscribe | 2234 |
| Параметры запроса | 2234 |
| Параметры ответа..... | 2234 |
| Команда login | 2238 |
| Параметры | 2238 |
| Ответ..... | 2239 |
| Команда logout | 2240 |
| Параметры запроса | 2241 |
| Команда profiles..... | 2241 |
| Параметры | 2242 |
| Ответ..... | 2242 |
| Call API..... | 2244 |
| Подразделы | 2244 |
| Call HTTP API..... | 2246 |

| | |
|---|------|
| Команды HTTP Call API | 2246 |
| HTTP Call API - answer.xsd | 2246 |
| HTTP Call API - get_commands_list.xsd | 2247 |
| HTTP Call API - login.xsd | 2248 |
| HTTP Call API - logout.xsd | 2249 |
| HTTP Call API - make_call.xsd | 2250 |
| HTTP Call API - monitor.xsd | 2251 |
| HTTP Call API - ping.xsd | 2253 |
| HTTP Call API - reject.xsd | 2253 |
| События HTTP Call API | 2254 |
| HTTP Call API - conversations_event.xsd | 2255 |
| API сервиса Автосекретарь..... | 2258 |
| Добавление слов в базу данных | 2258 |
| Таблица full names..... | 2258 |
| Добавление полных имен: | 2258 |
| Удаление полных имен: | 2259 |
| Таблица aliases..... | 2259 |
| Добавление алиаса: | 2259 |
| Удаление алиаса: | 2260 |
| Таблица hard_aliases (жесткие алиасы) | 2261 |
| Добавление..... | 2261 |
| Удаление | 2262 |
| Таблица extended_word | 2262 |
| Добавление..... | 2262 |
| Удаление | 2263 |
| Рекомпиляция ASR-модели..... | 2263 |

CSTA API

- [Поддерживаемые методы/события](#)

CSTA API позволяет управлять вызовами на ECSS-10, устанавливать новые вызовы, принимать/отбивать активные вызовы, управлять базовыми сервисами (HOLD/HOLE, CTR, ZWAY, CONF).

Команды для настройки API (параметры подключения) приведены в разделе [/api/ — команды настройки API CSTA](#).

Протокол CSTA на уровне ECSS-10 работает поверх TCP, как описано в ECMA 323 Annex J.2 (TCP without SOAP). Формат пакета запроса/ответа следующий:

```
| 0 1 | 2 3   | 4 5 6 7 | 8 . . . |
| 0 0 | Length | Invoke ID | XML message body |
```

Invoke ID :: string(), "0000" - "9999". Invoke ID = 9999 используется в случае посылки событий от ECSS-10 на клиента.

Поддерживаемые методы/события

1. Start application session request XML:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<StartApplicationSession xmlns="http://www.ecma-international.org/standards/ecma-354/
appl_session" xmlns:ns1="http://www.eltex.org/csta">
  <applicationInfo>
    <applicationID>Eltex ECSS-10 CSTA adapter</applicationID>
    <applicationSpecificInfo>
      <ns1:SessionLoginInfo xmlns:ns1="http://www.eltex.org/csta">
        <ns1:userName>p.city</ns1:userName>
        <ns1:password>1234</ns1:password>
      </ns1:SessionLoginInfo>
    </applicationSpecificInfo>
  </applicationInfo>
  <requestedProtocolVersions>
    <protocolVersion>http://www.ecma-international.org/standards/ecma-323/csta/ed3</
protocolVersion>
  </requestedProtocolVersions>
  <requestedSessionDuration>300</requestedSessionDuration>
</StartApplicationSession>
```

2. Start application session positive response XML:

```
<?xml version="1.0"?>
<StartApplicationSessionPosResponse xmlns="http://www.ecma-international.org/standards/
ecma-354/appl_session" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ecma-international.org/standards/ecma-354/appl_session
start-application-session.xsd">
  <sessionID>f76e5fe9c5e0d444</sessionID>
  <actualProtocolVersion>http://www.ecma-international.org/standards/ecma-323/csta/ed3</
actualProtocolVersion>
  <actualSessionDuration>300</actualSessionDuration>
</StartApplicationSessionPosResponse>
```

3. Prolonge application session request XML:

```
<?xml version="1.0"?>
<ResetApplicationSessionTimer xmlns="http://www.ecma-international.org/standards/ecma-354/
appl_session" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="htt
p://www.ecma-international.org/standards/ecma-354/appl_session reset-application-session-
timer.xsd">
  <sessionID>f76e5fe9c5e0d444</sessionID>
  <requestedSessionDuration>500</requestedSessionDuration>
</ResetApplicationSessionTimer>
```

4. Prolonge application session positive response XML:

```
<?xml version="1.0"?>
<ResetApplicationSessionTimerPosResponse xmlns="http://www.ecma-international.org/
standards/ecma-354/appl_session" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ecma-international.org/standards/ecma-354/appl_session
reset-application-session-timer.xsd">
  <actualSessionDuration>500</actualSessionDuration>
</ResetApplicationSessionTimerPosResponse>
```

5. Stop application session request XML:

```
<StopApplicationSession xmlns="http://www.ecma-international.org/standards/ecma-354/
appl_session">
  <sessionID>f76e5fe9c5e0d444</sessionID>
  <sessionEndReason>
    <definedEndReason>normal</definedEndReason>
  </sessionEndReason>
</StopApplicationSession>
```

6. Stop application session positive request XML:

```
<?xml version="1.0"?>
<StopApplicationSessionPosResponse xmlns="http://www.ecma-international.org/standards/
ecma-354/appl_session" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ecma-international.org/standards/ecma-354/appl_session stop-
application-session.xsd"/>
```

7. Subscribe on events from device with number 725 request XML:

```
<MonitorStart xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3">
  <monitorObject>
    <deviceObject typeOfNumber="dialingNumber">725</deviceObject>
  </monitorObject>
  <requestedMonitorFilter>
    <callcontrol>
      <bridged>true</bridged>
      <callCleared>false</callCleared>
      <conferenced>false</conferenced>
      <connectionCleared>false</connectionCleared>
      <delivered>false</delivered>
      <digitsDialed>true</digitsDialed>
      <diverted>true</diverted>
      <established>false</established>
      <failed>true</failed>
      <held>true</held>
      <networkCapabilitiesChanged>true</networkCapabilitiesChanged>
      <networkReached>true</networkReached>
      <offered>true</offered>
      <originated>false</originated>
      <queued>true</queued>
      <retrieved>true</retrieved>
      <serviceInitiated>false</serviceInitiated>
      <transferred>false</transferred>
    </callcontrol>
  </requestedMonitorFilter>
</MonitorStart>
```

8. Subscribe on events from device with number 725 response XML:

```
<?xml version="1.0"?>
<MonitorStartResponse xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.ecma-
international.org/standards/ecma-323/csta/ed3 monitor-start.xsd">
  <monitorCrossRefID>DMON:f76e605590f2bc03</monitorCrossRefID>
</MonitorStartResponse>
```

9. ServiceInitiatedEvent — out call from monitored number:


```

<?xml version="1.0"?>
<ServiceInitiatedEvent xmlns="http://www.ecma-international.org/standards/ecma-323/csta/
ed3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://
www.ecma-international.org/standards/ecma-323/csta/ed3 service-initiated-event.xsd">
  <monitorCrossRefID>DM0N:f76e9c86851a2adc</monitorCrossRefID>
  <initiatedConnection>
    <deviceID typeOfNumber="dialingNumber">725::MV80MTQ4Nzg5MjgwQDEwLjEwMC4xMDEuMjY=#Z
WNzc0kxNTEwNzk3ODIyMzA2MDQ1MDMxMDIw#MzE3MDY1Nzk5NA==</deviceID>
    <callID>f76e9d66720eb32d</callID>
  </initiatedConnection>
  <initiatingDevice>
    <deviceIdentifier typeOfNumber="dialingNumber" mediaClass="notKnown" bitRate="const
ant"
>725::MV80MTQ4Nzg5MjgwQDEwLjEwMC4xMDEuMjY=#ZWNzc0kxNTEwNzk3ODIyMzA2MDQ1MDMxMDIw#MzE3MDY1Nz
k5NA==</deviceIdentifier>
  </initiatingDevice>
  <calledDevice>
    <notKnown/>
  </calledDevice>
  <cause>newCall</cause>
</ServiceInitiatedEvent>

```

10. DeliveredEvent

```

<?xml version="1.0"?>
<DeliveredEvent xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.ecma-
international.org/standards/ecma-323/csta/ed3 delivered-event.xsd">
  <monitorCrossRefID>DM0N:f76e9c7a2075e497</monitorCrossRefID>
  <connection>
    <deviceID typeOfNumber="dialingNumber">722:::QkE6Zjc2ZT1kNjY3MjB1YjMyZDpmNzZlOWQ3MD
E2MwNmM2Q3#ZWNzc08xNTEwNzk3ODIyNDgyMTE2MDExODI3MTE=#MUZtNVFIVWwVXI1Yw==</deviceID>
    <callID>f76e9d66720eb32d</callID>
  </connection>
  <alertingDevice>
    <deviceIdentifier typeOfNumber="dialingNumber" mediaClass="notKnown" bitRate="const
ant"
>722:::QkE6Zjc2ZT1kNjY3MjB1YjMyZDpmNzZlOWQ3MDE2MwNmM2Q3#ZWNzc08xNTEwNzk3ODIyNDgyMTE2MDExODI
3MTE=#MUZtNVFIVWwVXI1Yw==</deviceIdentifier>
  </alertingDevice>
  <callingDevice>
    <deviceIdentifier typeOfNumber="dialingNumber" mediaClass="notKnown" bitRate="const
ant"
>725:::MV80MTQ4Nzg5MjgwQDEwLjEwMCA4xMDEuMjY=#ZWNzc08xNTEwNzk3ODIyMzA2MDQ1MDMxMDIw#MzE3MDY1Nz
k5NA==</deviceIdentifier>
  </callingDevice>
  <calledDevice>
    <deviceIdentifier typeOfNumber="dialingNumber" mediaClass="notKnown" bitRate="const
ant"
>722:::QkE6Zjc2ZT1kNjY3MjB1YjMyZDpmNzZlOWQ3MDE2MwNmM2Q3#ZWNzc08xNTEwNzk3ODIyNDgyMTE2MDExODI
3MTE=#MUZtNVFIVWwVXI1Yw==</deviceIdentifier>
  </calledDevice>
  <lastRedirectionDevice>
    <notKnown/>
  </lastRedirectionDevice>
  <cause>newCall</cause>
</DeliveredEvent>

```

11. EstablishedEvent

```

<?xml version="1.0"?>
<EstablishedEvent xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.ecma-
international.org/standards/ecma-323/csta/ed3 established-event.xsd">
  <monitorCrossRefID>DM0N:f76e9c7a2075e497</monitorCrossRefID>
  <establishedConnection>
    <deviceID typeOfNumber="dialingNumber">722:::QkE6Zjc2ZT1kNjY3MjB1YjMyZDpmNzZlOWQ3MD
E2MwNmM2Q3#ZWNzc08xNTEwNzk3ODIyNDgyMTE2MDExODI3MTE=#MUZtNVFIVWwVXI1Yw==</deviceID>
    <callID>f76e9d66720eb32d</callID>
  </establishedConnection>
  <answeringDevice>
    <deviceIdentifier typeOfNumber="dialingNumber" mediaClass="notKnown" bitRate="const
ant"
>722:::QkE6Zjc2ZT1kNjY3MjB1YjMyZDpmNzZlOWQ3MDE2MwNmM2Q3#ZWNzc08xNTEwNzk3ODIyNDgyMTE2MDExODI
3MTE=#MUZtNVFIVWwVXI1Yw==</deviceIdentifier>
  </answeringDevice>
  <callingDevice>
    <deviceIdentifier typeOfNumber="dialingNumber" mediaClass="notKnown" bitRate="const
ant"
>722:::QkE6Zjc2ZT1kNjY3MjB1YjMyZDpmNzZlOWQ3MDE2MwNmM2Q3#ZWNzc08xNTEwNzk3ODIyNDgyMTE2MDExODI
3MTE=#MUZtNVFIVWwVXI1Yw==</deviceIdentifier>
  </callingDevice>
  <calledDevice>
    <notKnown/>
  </calledDevice>
  <lastRedirectionDevice>
    <notKnown/>
  </lastRedirectionDevice>
  <cause>normal</cause>
</EstablishedEvent>

```

12. OriginatedEvent

```

<?xml version="1.0"?>
<OriginatedEvent xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.ecma-
international.org/standards/ecma-323/csta/ed3 originated-event.xsd">
  <monitorCrossRefID>DMON:f76e9c86851a2adc</monitorCrossRefID>
  <originatedConnection>
    <deviceID typeOfNumber="dialingNumber">725::MV80MTQ4Nzg5MjgwQDEwLjEwMC4xMDEuMjY=#Z
WNzc0kxNTEwNzk3ODIyMzA2MDQ1MDMxMDIw#MzE3MDY1Nzk5NA==</deviceID>
    <callID>f76e9d66720eb32d</callID>
  </originatedConnection>
  <callingDevice>
    <deviceIdentifier typeOfNumber="dialingNumber" mediaClass="notKnown" bitRate="const
ant"
>725::MV80MTQ4Nzg5MjgwQDEwLjEwMC4xMDEuMjY=#ZWNzc0kxNTEwNzk3ODIyMzA2MDQ1MDMxMDIw#MzE3MDY1Nz
k5NA==</deviceIdentifier>
  </callingDevice>
  <calledDevice>
    <notKnown/>
  </calledDevice>
  <localConnectionInfo>connected</localConnectionInfo>
  <cause>normal</cause>
</OriginatedEvent>

```

13. ConnectionClearedEvent

```

<?xml version="1.0"?>
<ConnectionClearedEvent xmlns="http://www.ecma-international.org/standards/ecma-323/csta/
ed3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://
www.ecma-international.org/standards/ecma-323/csta/ed3 connection-cleared-event.xsd">
  <monitorCrossRefID>DMON:f76e9c7a2075e497</monitorCrossRefID>
  <droppedConnection>
    <deviceID typeOfNumber="dialingNumber">722::QkE6Zjc2ZTlkNjY3MjB1YjMyZDpmNzZlOWQ3MD
E2MwNmM2Q3#ZWNzc08xNTEwNzk3ODIyNDgyMTE2MDExODI3MTE=#MUZtNVFIVWVwVXI1YW==</deviceID>
    <callID>f76e9d66720eb32d</callID>
  </droppedConnection>
  <releasingDevice>
    <deviceIdentifier typeOfNumber="dialingNumber" mediaClass="notKnown" bitRate="const
ant"
>722::QkE6Zjc2ZTlkNjY3MjB1YjMyZDpmNzZlOWQ3MDE2MwNmM2Q3#ZWNzc08xNTEwNzk3ODIyNDgyMTE2MDExODI
3MTE=#MUZtNVFIVWVwVXI1YW==</deviceIdentifier>
  </releasingDevice>
  <cause>normalClearing</cause>
  <extensions>
    <privateData>
      <string>release_initiator=system;</string>
    </privateData>
  </extensions>
</ConnectionClearedEvent>

```

14. Make out call request XML:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<MakeCall xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3">
  <callingDevice typeOfNumber="dialingNumber">725</callingDevice>
  <calledDirectoryNumber typeOfNumber="dialingNumber">722</calledDirectoryNumber>
</MakeCall>
```

15. Make out call response XML:

```
<?xml version="1.0"?>
<MakeCallResponse xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.ecma-
international.org/standards/ecma-323/csta/ed3 make-call.xsd">
  <callingDevice>
    <deviceID typeOfNumber="dialingNumber">725:::QkE6Zjc2ZjI1NDBlYmU5MGY0NjpmNzZmMjU0Mj
UwYTAzODll#ZWNzc08xNTEwNzk4NDA1ODI3ODI4MDE2MTE2MTE=#MTEyMzg5MDIxMQ==</deviceID>
    <callID>f76f2540ebe90f46</callID>
  </callingDevice>
</MakeCallResponse>
```

16. Hold call request XML:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<HoldCall xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3">
  <callToBeHeld>
    <deviceID typeOfNumber="dialingNumber">725:::QkE6Zjc2ZjI1NDBlYmU5MGY0NjpmNzZmMjU0Mj
UwYTAzODll#ZWNzc08xNTEwNzk4NDA1ODI3ODI4MDE2MTE2MTE=#MTEyMzg5MDIxMQ==</deviceID>
    <callID>f76f2540ebe90f46</callID>
  </callToBeHeld>
  <connectionReservation>>false</connectionReservation>
</HoldCall>
```

17. Hold call response XML:

```
<?xml version="1.0"?>
<HoldCallResponse xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.ecma-
international.org/standards/ecma-323/csta/ed3 hold-call.xsd"/>
```

18. UnHold call request XML:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<RetrieveCall xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3">
  <callToBeRetrieved>
    <deviceID typeOfNumber="dialingNumber">725::QkE6Zjc2ZjI1NDBlYmU5MGY0NjpmNzZmMjU0Mj
UwYTAzODll#ZWNzc08xNTEwNzk4NDA1ODI3ODI4MDE2MTE2MTE=#MTEyMzg5MDIxMQ==</deviceID>
    <callID>f76f2540ebe90f46</callID>
  </callToBeRetrieved>
</RetrieveCall>>
```

19. UnHold call response XML:

```
<?xml version="1.0"?>
<RetrieveCallResponse xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.ecma-
international.org/standards/ecma-323/csta/ed3 retrieve-call.xsd"/>
```

20. Clear connection request XML:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ClearConnection xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3">
  <connectionToBeCleared>
    <deviceID typeOfNumber="dialingNumber">722::QkE6Zjc2ZjUzZDlhODk3NzIyODpmNzZmNTNlM2
QzMjVmNGJh#ZWNzc08xNTEwNzk4NjA2MTEwMDUzMDE3NTg2MTE=#MzVhbnllOXJ5MTgzZQ==</deviceID>
    <callID>f76f53d9a8977228</callID>
  </connectionToBeCleared>
</ClearConnection>
```

21. Clear connection response XML:

```
<?xml version="1.0"?>
<ClearConnectionResponse xmlns="http://www.ecma-international.org/standards/ecma-323/csta/
ed3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://
www.ecma-international.org/standards/ecma-323/csta/ed3 clear-connection.xsd"/>
```

22. Clear call request XML:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ClearCall xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3">
  <callToBeCleared>
    <deviceID typeOfNumber="dialingNumber">722::QkE6Zjc2ZjY1OWYzZWx0GY1NTpmNzZmNjVhOT
c2MDhiZjU1#ZWNzc08xNTEwNzk4NjgyNDM5NTg0MDE4MTEyMTE=#akQ2RlM0NURLN3B2Yw==</deviceID>
    <callID>f76f659f3ec18f55</callID>
  </callToBeCleared>
</ClearCall>
```

23. Clear call response XML:

```
<?xml version="1.0"?>
<ClearCallResponse xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.ecma-
international.org/standards/ecma-323/csta/ed3 clear-call.xsd"/>
```

24. Get calls on device request XML:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<SnapshotDevice xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3">
  <snapshotObject typeOfNumber="dialingNumber">725</snapshotObject>
</SnapshotDevice>
```

25. Get calls on device response XML:

```
<?xml version="1.0"?>
<SnapshotDeviceResponse xmlns="http://www.ecma-international.org/standards/ecma-323/csta/
ed3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://
www.ecma-international.org/standards/ecma-323/csta/ed3 snapshot-device.xsd">
  <crossRefIDorSnapshotData>
    <snapshotData>
      <snapshotDeviceResponseInfo>
        <connectionIdentifier>
          <deviceID typeOfNumber="dialingNumber">725::MV8xNDY1Mzg4MjgyQDEwLjEwMC
4xMDEuMjY=#ZWZzc0kxNTEwNzk4Nzk5MTAwODI5MDMxMDMw#NDAzODU3NTA5OQ==</deviceID>
          <callID>f76f7f32a8b2112e</callID>
        </connectionIdentifier>
        <localCallState>
          <compoundCallState>
            <localConnectionState>alerting</localConnectionState>
            <localConnectionState>alerting</localConnectionState>
          </compoundCallState>
        </localCallState>
      </snapshotDeviceResponseInfo>
    </snapshotData>
  </crossRefIDorSnapshotData>
</SnapshotDeviceResponse>
```

26. Answer call request XML:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<AnswerCall xmlns="http://www.ecma-international.org/standards/ecma-323/csta/ed3">
  <callToBeAnswered>
    <deviceID typeOfNumber="dialingNumber">725::QkE6Zjc2Zjk4MDM4NjBjMjZkZjpmNzZmOTgwZT
ZmMzk4ZDgz#ZWNzc08xNTEwNzk4ODk4ODgzOTA1MDE5NzQ5MTE=#MzEzMjc2MjM0NA==</deviceID>
    <callID>f76f9803860c26df</callID>
  </callToBeAnswered>
</AnswerCall>
```

27. Answer call response XML:

```
<?xml version="1.0"?><AnswerCallResponse xmlns="http://www.ecma-international.org/
standards/ecma-323/csta/ed3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ecma-international.org/standards/ecma-323/csta/ed3 answer-
call.xsd"/>
```

28. Transfer call request XML:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<SingleStepTransferCall xmlns="http://www.ecma-international.org/standards/ecma-323/csta/
ed3">
  <activeCall>
    <deviceID typeOfNumber="dialingNumber">725::QkE6Zjc2ZmI2ZDRhNjViNzE0NzpmNzZmYjZkNj
ZmNjY2YTRi#ZWNzc08xNTEwNzk5MDMxMDgyOTg4MDIwNzYzMTE=#MzUyODQwMTIyNg==</deviceID>
    <callID>f76fb6d4a65b7147</callID>
  </activeCall>
  <transferredTo typeOfNumber="dialingNumber">723</transferredTo>
</SingleStepTransferCall>
```

29. Transfer call response XML:

```
<?xml version="1.0"?> <SingleStepTransferCallResponse xmlns="http://www.ecma-
international.org/standards/ecma-323/csta/ed3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://www.ecma-international.org/standards/ecma-323/csta/ed3
single-step-transfer.xsd"> <transferredCall> <deviceID typeOfNumber="dialingNumber">723::Q
kE6Zjc2ZmI2ZDRhNjViNzE0NzpmNzZmYjZk4ODI4MGU1MmY2#ZWNzc08xNTEwNzk5MDM0MDY4MDEyMDIwNzZk4MTE=#S3
JIVWMzY0tRYVpVRA==</deviceID> <callID>f76fb6d4a65b7147</callID> </transferredCall> </
SingleStepTransferCallResponse>
```


API по управлению конфигурацией ECSS-10 (http-terminal)

В данном разделе приведено описание команд для управления системой ECSS-10 через HTTP-терминал.

Управление любым телекоммуникационным оборудованием включает в себе множество различных задач и аспектов, часть задач выполняют операторы посредством командной консоли (CoCon), другие задачи с помощью Web-интерфейса.

Есть ряд телекоммуникационных задач, для выполнения которых по ряду причин не подходит командная консоль CoCon или Web-интерфейс, например, создание абонентов. При большом количестве абонентов в короткий промежуток времени необходимо выполнить множество однотипных команд с разными данными. Для решения подобных задач эффективней применять различные скрипты и средства автоматизации. Если оператор хочет реализовать свой интерфейс управления комплексом, например, рабочее место менеджера по работе с клиентами, осуществляющего заключение договора с клиентом и добавление новых абонентов в систему.



- HTTP-терминал является одним из возможных вариантов предоставления интерфейса управления системой ECSS-10.
- HTTP-терминал — это открытое API, позволяющее предоставить оператору интерфейс управления комплексом ECSS-10 с помощью web-сервисов (выполнение запросов по протоколу HTTP/HTTPS).
- API (Application Programming Interface) — это интерфейс, описывающий необходимые параметры и алгоритм для взаимодействия между различными модулями программ.

Сессионная работа HTTP-терминала

Описание работы HTTP-терминала в общем случае:

- 1. Установление сессии
- 2. Выполнение команд, периодические ping-запросы
- 3. Завершение сессии
- 4. Пример кода на Java для работы с HTTP-терминалом

1. Установление сессии

Для создания сессии нужно отправить POST-запрос командой `http(s)://<ServerIP>:<ServerPort>/system/login` вида:

```
<in>
  <login user = "ssw" password="ssw" />
</in>
```

Результатом этого запроса в случае успеха будет код 200 и token сессии (в cookie), иначе — код 404.

После установления сессии во всех последующих запросах в рамках данной сессии необходимо пробрасывать в *cookie* полученный *token*.

В случае успешной авторизации в теле ответа придет "xml", содержащая мета-информацию, которая может потребоваться в последующих запросах:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="login.xsd"
">
  <settings md="md1" ds="ds1" version="3.10.0.36"/>
</out>
```

2. Выполнение команд, периодические ping-запросы

HTTP-терминал позволяет отправлять по http в рамках одной сессии несколько параллельных запросов. Для того чтобы отличать ответы на данные запросы, в заголовке запросов можно разместить параметр "P-Request-Id", значение которого будет протранслировано в ответе.

Если запрос к HTTP-терминалу был выполнен без указания token-а в cookie либо сессии для данного token-а уже нет, в ответ на такой запрос будет отправлена ошибка "401". Получив такую ошибку, необходимо заново пройти процесс аутентификации в системе.

Для того чтобы проверить, состояние сессии с определенным token-ом, используется команда:

```
http(s)://<ServerIP>:<ServerPort>/system/is_active
```

Для данной команды необходимо передать только *token* в *cookie* запроса методом GET.

Если клиент долгое время (60 минут) не выполняет никаких запросов с данным token-ом, то сессия на стороне HTTP-терминала автоматически закрывается. Чтобы продлить время жизни сессии со стороны клиента, должны приходить периодические запросы методом *post*:

```
http(s)://<ServerIP>:<ServerPort>/system/ping
```

Запросы должны содержать *token* в *cookie* запроса. В случае успешного продления сессии возвращается код 200. Если сессия с данным *token* не найдена — вернется 404.

3. Завершение сессии

Чтобы завершить сессию со стороны клиента, необходимо на сервер отправить команду:

`http(s)://<ServerIP>:<ServerPort>/system/logout`

методом GET с *token* в *cookie* запроса.

В случае успешного *logout* придет ответ 200. Если сессия не найдена — вернется 404.

4. Пример кода на Java для работы с HTTP-терминалом

```

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.client.CookieStore;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.cookie.Cookie;
import org.apache.http.entity.ByteArrayEntity;
import org.apache.http.impl.client.BasicCookieStore;
import org.apache.http.impl.client.CloseableHttpClient;
import org.apache.http.impl.client.HttpClients;
import org.apache.http.impl.cookie.BasicClientCookie;
import org.apache.http.util.EntityUtils;

import java.io.IOException;

/**
 * Elemental example for executing multiple POST requests sequentially.
 */
public class Main {

    public static void main(String[] args) throws Exception {
        String host = "http://192.168.23.87:9999";
        CookieStore cookie = login("ssw", "ssw", host);
        if (isActive(cookie, host))
        {
            uptime(cookie, host);
            logout(cookie, host);
        }
    }

    private static CookieStore login(String login, String password, String host) throws
IOException
    {
        CookieStore cookieStore = new BasicCookieStore();
        CloseableHttpClient client = HttpClients.custom()
            .setDefaultCookieStore(cookieStore)
            .build();
        HttpPost httpPost = new HttpPost(host + "/system/login");
        httpPost.setHeader("Content-Type", "text/xml");
        String xml = "<in><login user = \" + login + "\" password=\" + password + "\" /></in>";
        HttpEntity entity = new ByteArrayEntity(xml.getBytes("UTF-8"));
        httpPost.setEntity(entity);
        HttpResponse response = client.execute(httpPost);
        String loginResult = EntityUtils.toString(response.getEntity());
        int loginResultCode = response.getStatusLine().getStatusCode();
    }
}

```

```

System.out.println("Login rc " + loginResultCode);
System.out.println("Login result " + loginResult);
CookieStore resultCookieStore = new BasicCookieStore();
for (Cookie c : cookieStore.getCookies())
{
    if (c.getName().equals("token") && c instanceof BasicClientCookie)
    {
        // Нужно для того, чтобы token применялся для всех
        ((BasicClientCookie)c).setPath("/");
        resultCookieStore.addCookie(c);
        return resultCookieStore;
    }
}
return cookieStore;
}

private static boolean isActive(CookieStore cookieStore, String host) throws IOException
{
    CloseableHttpClient client = HttpClients.custom()
        .setDefaultCookieStore(cookieStore)
        .build();
    HttpGet httpGet = new HttpGet(host + "/system/is_active");
    HttpResponse response = client.execute(httpGet);
    int isActiveResultCode = response.getStatusLine().getStatusCode();
    System.out.println("Is Active rc " + isActiveResultCode);
    return isActiveResultCode == 200;
}

private static void uptime(CookieStore cookieStore, String host) throws IOException
{
    CloseableHttpClient client = HttpClients.custom()
        .setDefaultCookieStore(cookieStore)
        .build();
    HttpPost httpPost = new HttpPost(host + "/commands/get_info");
    httpPost.setHeader("Content-Type", "text/xml");
    String xml = "<in xmlns:xs=\"http://www.w3.org/2001/XMLSchema-instance\"
xs:noNamespaceSchemaLocation=\"get_info.xsd\" />";
    HttpEntity entity = new ByteArrayEntity(xml.getBytes("UTF-8"));
    httpPost.setEntity(entity);
    HttpResponse response = client.execute(httpPost);
    String uptimeResult = EntityUtils.toString(response.getEntity());
    int isActiveResultCode = response.getStatusLine().getStatusCode();
    System.out.println("Uptime rc " + isActiveResultCode);
    System.out.println("Uptime result " + uptimeResult);
}

private static void logout(CookieStore cookieStore, String host) throws IOException
{
    CloseableHttpClient client = HttpClients.custom()
        .setDefaultCookieStore(cookieStore)

```

```

        .build();
        HttpGet httpGet = new HttpGet(host + "/system/logout");
        HttpResponse response = client.execute(httpGet);
        int isActiveResultCode = response.getStatusLine().getStatusCode();
        System.out.println("Logout rc " + isActiveResultCode);
    }
}

```

Для проверки работы команд с различными параметрами есть тестовая страничка по адресу: <IP_Web-conf>/cluster/xml_request/. Предварительно в соседней вкладке браузера нужно залогиниться в web-конфигуратор.

Управление HTTP-терминалом

Подразделы

Нс_aaa_access_options - Управление подсистемой RADIUS авторизации

- [Нс_aaa_access_options_clean](#)
- [Нс_aaa_access_options_info](#)
- [Нс_aaa_access_options_set](#)

[Нс_aaa_access_options_clean](#)

Команда сброса настроек подсистемы RADIUS авторизации (access).

Аналог команды в CoSoп:

domain/<DOMAIN>/aaa/access/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/aaa_access_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean properties of AAA service.
      Usage: http://server:9999/commands/aaa_access_options_clean.
      Implementation: hc_aaa_access_options_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/aaa_access_options_clean

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_aaa_access_options_clean.xsd">
  <request>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <option name="enable"/>
        <option name="my_address"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_aaa_access_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="enable"/>
        <success option="my_address"/>
      </domain>
    </domain_options>
  </result>
</out>

```

[Hc_aaa_access_options_info](#)

Команда просмотра информации о подсистеме RADIUS авторизации (access).

Аналог команды в CoCon:

domain/<DOMAIN>/aaa/access/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/aaa_access_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of AAA service.
      Usage: http://server:9999/commands/aaa\_access\_options\_info.
      Implementation: hc_aaa_access_options_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- Types -->
  <xs:simpleType name="resendCountType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="50"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="aaaAccessOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="enable" type="booleanType" use="optional"/>
        <xs:attribute name="my_address" type="ipAddressType" use="optional"/>
        <xs:attribute name="servers" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              The list of the RADIUS servers' names splitted by comma.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="login" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Access RADIUS request login.
              Available values: default | billing | sip | cgpn | any string.
              In case of login = default - login set as the domain name.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="password" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Access RADIUS request password.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

Available values: default | billing | sip | any string.
In case of password = default - password set as the domain name.
In case of strong security restrictions password may be shown as
*****
        </xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoInType" minOccurs="
1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/aaa_access_options_info

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_aaa_access_options_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_aaa_access_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="aaaAccessOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="aaaAccessOptionsType" servers="billing_access"
password="*****" my_address="192.168.1.21" login="default" enable="false"/>
      </domain>
    </domain_options>
  </result>
</out>
```

hc_aaa_access_options_set

Команда изменения настроек подсистемы RADIUS авторизации (access).

Аналог команды в CoCon:

domain/<DOMAIN>/aaa/access/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/aaa_access_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set AAA service properties
      Usage: http://server:9999/commands/aaa_access_options_set.
      Implementation: hc_aaa_access_options_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- Types -->
  <xs:simpleType name="resendCountType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="50"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="aaaAccessOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="enable" type="booleanType" use="optional"/>
        <xs:attribute name="my_address" type="ipAddressType" use="optional"/>
        <xs:attribute name="servers" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              The list of the RADIUS servers' names splitted by comma.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="login" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Access RADIUS request login.
              Available values: default | billing | sip | cgpn | any string.
              In case of login = default - login set as the domain name.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="password" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Access RADIUS request password.
              Available values: default | billing | sip | any string.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

                In case of password = default - password set as the domain name.
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
    <xs:all>
        <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainSetInType"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
            <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/aaa_access_options_set

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_aaa_access_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="aaaAccessOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="aaaAccessOptionsType" servers="master,192.168.2.21"
my_address="192.168.1.21" enable="false"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_aaa_access_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="enable"/>
        <success option="my_address"/>
        <success option="servers"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Hc_aaa_general_options - Управление общими настройками службы RADIUS AAA

- [Hc_aaa_general_options_clean](#)
- [Hc_aaa_general_options_info](#)
- [Hc_aaa_general_options_set](#)

Ис_aaa_general_options_clean

Команда сброса общих настроек подсистемы аккаунтинга.

Аналог команды в CoCon:

`domain/<DOMAIN>/aaa/general/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/aaa_general_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean properties of AAA service.
      Usage: http://server:9999/commands/aaa\_general\_options\_clean.
      Implementation: hc_aaa_general_options_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2020, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Список всех возможных атрибутов аналогичен Hc_aaa_general_options_info и Hc_aaa_general_options_set.

Пример:

Запрос:

http://192.168.1.21:9999/commands/aaa_general_options_clean

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_aaa_general_options_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="adaptation" />
        <option name="call_origin_mode" />
        <option name="if_radius_unavailable" />
        <option name="modificate_station_id" />
        <option name="profile" />
      </domain>
    </domain_options>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_aaa_general_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="adaptation"/>
        <success option="call_origin_mode"/>
        <success option="if_radius_unavailable"/>
        <success option="modificate_station_id"/>
        <success option="profile"/>
      </domain>
    </domain_options>
  </result>
</out>
```

[Hc_aaa_general_options_info](#)

Команда просмотра информации об общих настройках подсистемы аккаунтинга.

Аналог команды в CoCon:

domain/<DOMAIN>/aaa/general/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/aaa_general_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of AAA service.
      Usage: http://server:9999/commands/aaa\_general\_options\_info.
      Implementation: hc_aaa_general_options_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2020, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- Types -->
  <xs:simpleType name="resendCountType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="50"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- callOriginModeType -->
  <xs:simpleType name="callOriginModeType">
    <xs:restriction base="unionType">
      <xs:enumeration value="default"/>
      <xs:enumeration value="cisco"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="aaaGeneralOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="adaptation" type="atomType" use="optional"/>
        <xs:attribute name="if_radius_unavailable" type="xs:string" use="optional"/>
        <xs:attribute name="profile" type="xs:string" use="optional"/>
        <xs:attribute name="call_origin_mode" type="callOriginModeType" use="optional"/>
        <xs:attribute name="modificate_station_id" type="booleanType" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>

```

```

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoInType" minOccurs="
1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/aaa_general_options_info

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_aaa_general_options_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_aaa_general_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="aaaGeneralOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="aaaGeneralOptionsType" profile="cisco_vsa"
        modificate_station_id="true" if_radius_unavailable="private,local" call_origin_mode="cisco"
        adaptation="default_adaptation"/>
      </domain>
    </domain_options>
  </result>
</out>
```

hc_aaa_general_options_set

Команда изменения общих настроек подсистемы аккаунтинга.

Аналог команды в CoCon:

domain/<DOMAIN>/aaa/general/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/aaa_general_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set AAA service properties
      Usage: http://server:9999/commands/aaa_general_options_set.
      Implementation: hc_aaa_general_options_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2020, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- Types -->
  <xs:simpleType name="resendCountType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="50"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- callOriginModeType -->
  <xs:simpleType name="callOriginModeType">
    <xs:restriction base="unionType">
      <xs:enumeration value="default"/>
      <xs:enumeration value="cisco"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="aaaGeneralOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="adaptation" type="atomType" use="optional"/>
        <xs:attribute name="if_radius_unavailable" type="xs:string" use="optional"/>
        <xs:attribute name="profile" type="xs:string" use="optional"/>
        <xs:attribute name="call_origin_mode" type="callOriginModeType" use="optional"/>
        <xs:attribute name="modificate_station_id" type="booleanType" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->

```



```

<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/aaa_general_options_set

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_aaa_general_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="aaaGeneralOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="aaaGeneralOptionsType" adaptation="default_adaptation"
call_origin_mode="cisco" if_radius_unavailable="private, local" modificate_station_id="true"
profile="cisco_vsa" />
      </domain>
    </domain_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_aaa_general_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="adaptation"/>
        <success option="call_origin_mode"/>
        <success option="if_radius_unavailable"/>
        <success option="modificate_station_id"/>
        <success option="profile"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_aaa_options - Управление службой RADIUS AAA

- [Нс_aaa_options_clean](#)
- [Нс_aaa_options_info](#)
- [Нс_aaa_options_set](#)

Нс_aaa_options_clean

Команда сброса настроек подсистемы аккаунтинга.

Аналог команды в CoCon:

domain/<DOMAIN>/aaa/accounting/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/aaa_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean properties of AAA service.
      Usage: http://server:9999/commands/aaa\_options\_clean.
      Implementation: hc_aaa_options_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Список всех возможных атрибутов аналогичен Hc_aaa_options_info и Hc_aaa_options_set.

Пример:

Запрос:

http://192.168.1.21:9999/commands/aaa_options_clean

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_aaa_options_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="send_ss_notification" />
        <option name="my_address" />
        <option name="enable" />
        <option name="interim_interval" />
        <option name="unsuccessful_call_info" />
        <option name="servers" />
      </domain>
    </domain_options>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_aaa_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="enable"/>
        <success option="interim_interval"/>
        <success option="my_address"/>
        <success option="send_ss_notification"/>
        <success option="servers"/>
        <success option="unsuccessful_call_info"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Нс_aaa_options_info

Команда просмотра информации о подсистеме аккаунтинга.

Аналог команды в CoCon:

domain/<DOMAIN>/aaa/accounting/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/aaa_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of AAA service.
      Usage: http://server:9999/commands/aaa\_options\_info.
      Implementation: hc_aaa_options_info.erl.
      Git: git@git.eltex.loc:8080/httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- Types -->
  <xs:simpleType name="resendCountType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="50"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="aaaOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="send_ss_notification" type="booleanType" use="optional"/>
        <xs:attribute name="enable" type="booleanType" use="optional"/>
        <xs:attribute name="my_address" type="ipAddressType" use="optional"/>
        <xs:attribute name="servers" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              The list of the RADIUS servers' names splitted by comma.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="interim_interval" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Acct-Interim-Interval value. Available values:
              server_configured - Acct-Interim-Interval value get from RADIUS
              accept message;
              disabled - account update messages doesn't send;
              [60, 86400] - Acct-Interim-Interval value in seconds.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="unsuccessful_call_info" type="booleanType" use="optional"/>

```

```

        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
    <xs:all>
        <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainInfoInType" minOccurs="
1"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/aaa_options_info

```

<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_aaa_options_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_aaa_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="aaaOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="aaaOptionsType" unsuccessful_call_info="false"
servers="billing_account,billing_access" send_ss_notification="false" my_address="192.168.1.21"
interim_interval="server_configured" enable="false"/>
      </domain>
    </domain_options>
  </result>
</out>
```

hc_aaa_options_set

Команда изменения настроек подсистемы аккаунтинга.

Аналог команды в CoCon:

domain/<DOMAIN>/aaa/accounting/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/aaa_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set AAA service properties
      Usage: http://server:9999/commands/aaa_options_set.
      Implementation: hc_aaa_options_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- Types -->
  <xs:simpleType name="resendCountType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="50"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="aaaOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="send_ss_notification" type="booleanType" use="optional"/>
        <xs:attribute name="enable" type="booleanType" use="optional"/>
        <xs:attribute name="my_address" type="ipAddressType" use="optional"/>
        <xs:attribute name="servers" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              The list of the RADIUS servers' names splitted by comma.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="interim_interval" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Acct-Interim-Interval value. Available values:
              server_configured - Acct-Interim-Interval value get from RADIUS
              accept message;
              disabled - account update messages doesn't send;
              [60, 86400] - Acct-Interim-Interval value in seconds.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="unsuccessful_call_info" type="booleanType" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

        </xs:complexContent>
    </xs:complexType>
    <!-- resultType-->
    <xs:complexType name="resultType">
        <xs:all>
            <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
        </xs:all>
    </xs:complexType>
    <!-- in-->
    <xs:element name="in">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="request">
                    <xs:complexType>
                        <xs:all>
                            <xs:element name="domain_options" type="domainSetInType"/>
                        </xs:all>
                    </xs:complexType>
                </xs:element>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <!-- out -->
    <xs:element name="out">
        <xs:complexType>
            <xs:choice>
                <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
                <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
            </xs:choice>
        </xs:complexType>
    </xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/aaa_options_set

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_aaa_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="aaaOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="aaaOptionsType" servers="master,naomi2" my_address="127.0.0.1"
        unsuccessful_call_info="false" send_ss_notification="false" enable="false"
        interim_interval="server_configured" />
      </domain>
    </domain_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_aaa_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="enable"/>
        <success option="interim_interval"/>
        <success option="my_address"/>
        <success option="send_ss_notification"/>
        <success option="servers"/>
        <success option="unsuccessful_call_info"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Hc_adaptation_context - Управление контекстами адаптации

- [Hc_adaptation_context_delete](#)
- [Hc_adaptation_context_import](#)
- [Hc_adaptation_context_list](#)

- [Hc_adaptation_context_show](#)

[Hc_adaptation_context_delete](#)

Команда предназначена для удаления контекста адаптации номеров по имени из определенного домена.

Аналог команды в CoCon:

`cluster/storage/<STORAGE>/adaptation/delete`

`domain/<DOMAIN>/adaptation/delete`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/adaptation_context_delete

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="context" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="error" type="errorType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1

Удаление существующего контекста

Запрос:

http://192.168.1.21:9999/commands/adaptation_context_delete

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_adaptation_context_delete.xsd">
  <request domain="biysk.local" context="mod1_adp"/>
</in>

```

Ответ:

204

Пример 2

Удаление несуществующего контекста

Запрос:

http://192.168.1.21:9999/commands/adaptation_context_delete

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_adaptation_context_delete.xsd">
  <request domain="biysk.local" context="adp2"/>
</in>
```

Ответ:

404

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_adaptat
  ion_context_delete.xsd">
  <error cmd="adaptation_context_delete" reason="not_exists" body="not_found" entity="adp2"/>
</out>
```

hc_adaptation_context_import

Команда предназначена для импорта контекста адаптации номеров в определенный домен. Если контекст в указанном имени уже существует - контекст будет обновлен.

Аналог команды в CoCon:

cluster/storage/<STORAGE>/adaptation/import

domain/<DOMAIN>/adaptation/import

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/adaptation_context_import

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />
  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- progressType -->
  <xs:complexType name="progressType">
    <xs:sequence>
      <xs:element name="ref" minOccurs="1" maxOccurs="1" type="valueType" />
      <xs:element name="text" minOccurs="1" maxOccurs="1" type="valueType" />
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="context" type="xs:base64Binary" />
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required" />
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="error" type="errorType" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- progress -->
  <xs:element name="progress" type="progressType"/>
</xs:schema>

```

Пример 1:

Импорт валидного контекста

Запрос:

http://192.168.1.21:9999/commands/adaptation_context_import

```
<in>
  <request domain="biysk.local">
    <context>PD94bWwgdMvYc2lVbj0iMS4wIiB1bmNvZGluZz0iVVRGLTgiPz48YWRhcHRhdG1vbiB4bWxuczp4cz0i
aHR0cDovL3d3dy53My5vcmcvMjAwMS9YTUxTY2h1bWEtaW5zdGFuY2UiIHhzOm5vTmFtZXNwYWNlU2NoZW1hTG9jYXRpb249I
mVjc3NfbW9kawZpY2F0b3JzLnhzZCIgZmFtZT0iYWRwX3NvcM0iPjxydWx1IG5hbWU9ImZvc19zb3JtIj48Y29uZG10aw9ucy
8+PGFjdG1vbnMvPjxyZXN1bHQ+PGZpbm1zaC8+PC9yZXN1bHQ+PC9ydWx1PjwvYWRhcHRhdG1vbj4=</context>
  </request>
</in>
```

Ответ:

201

Пример 2

Запрос невалидного контекста

Запрос:

http://192.168.1.21:9999/commands/adaptation_context_import

```
<in>
<request domain="biysk.local">
<context
>PD94bWwgdMvYc2lVbj0iMS4wIiB1bmNvZGluZz0iVVRGLTgiPz48YWRhcHRhdB4bWxuczp4cz0iaHR0cDovL3d3dy53My5vc
mcvMjAwMS9YTUxTY2h1bWEtaW5zdGFuY2UiIHhzOm5vTmFtZXNwYWNlU2NoZW1hTG9jYXRpb249ImVjc3NfbW9kawZpY2F0b3
JzLnhzZCIgZmFtZT0iYWRwX3NvcM0iPjxydWx1IG5hbWU9ImZvc19zb3JtIj48Y29uZG10aw9ucy8+PGFjdG1vbnMvPjxyZXN
1bHQ+PGZpbm1zaC8+PC9yZXN1bHQ+PC9ydWx1PjwvYWRhcHRhdG1vbj4=</context>
</request>
</in>
```

Ответ:

404

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_adaptat
ion_context_import.xsd">
  <error cmd="adaptation_context_import" reason="invalid_input_xml" body="{invalid_xml, [{"Parse
xml error: {1,xmerl_b64Bin,[\"syntax error before: \",[\"\\\\\"=\\\\\\\"}]}}]" entity="<?xml version='
1.0' encoding='latin1'?> <in> <request domain="biysk.local"> <context>PD94bWwgdMvYc2lVbj0iMS4wIiB
1bmNvZGluZz0iVVRGLTgiPz48YWRhcHRhdB4bWxuczp4cz0iaHR0cDovL3d3dy53My5vcmcvMjAwMS9YTUxTY2h1bWEtaW5zd
GFuY2UiIHhzOm5vTmFtZXNwYWNlU2NoZW1hTG9jYXRpb249ImVjc3NfbW9kawZpY2F0b3JzLnhzZCIgZmFtZT0iYWRwX3Nvc
M0iPjxydWx1IG5hbWU9ImZvc19zb3JtIj48Y29uZG10aw9ucy8+PGFjdG1vbnMvPjxyZXN1bHQ+PGZpbm1zaC8+PC9yZXN1bHQ
+PC9ydWx1PjwvYWRhcHRhdG1vbj4=</context> </request> </in>" />
</out>
```


`hc_adaptation_context_list`

Команда предназначена для просмотра списка контекстов адаптации номеров в определенном домене.

Аналог команды в CoCon:

`cluster/storage/<STORAGE>/adaptation/list`

`domain/<DOMAIN>/adaptation/list`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/adaptation_context_list`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- contextType-->
  <xs:complexType name="contextType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="contexts">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="context" type="contextType" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/adaptation_context_list

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_adaptation_context_list.xsd">
  <request domain="biysk.local"/>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_adaptat
ion_context_list.xsd">
  <contexts>
    <context name="mod1_adp"/>
    <context name="adp_sorm"/>
    <context name="default_adaptation"/>
  </contexts>
</out>

```

hc_adaptation_context_show

Команда предназначена для получения контекста адаптации номеров по имени из определенного домена.

Аналог команды в CoCon:

cluster/storage/<STORAGE>/adaptation/show

domain/<DOMAIN>/adaptation/show

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/adaptation_context_show

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="context" type="xs:base64Binary"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="context" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

В поле context в результате в кодировке base64 закодирован контекст адаптации (в формате XML). XSD схема, валидирующая контексты адаптации:

ecss_modifiers.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- categoryType -->
  <xs:simpleType name="categoryType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="unknownAtThisTime"/>
      <xs:enumeration value="operatorFrench"/>
      <xs:enumeration value="operatorEnglish">
        <xs:annotation>
          <xs:documentation xml:lang="en">Obsolete value. Correct value is
operatorEnglish</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="operatorEnglish"/>
      <xs:enumeration value="operatorGerman"/>
      <xs:enumeration value="operatorRussian"/>
      <xs:enumeration value="operatorSpanish"/>
      <xs:enumeration value="reserved"/>
      <xs:enumeration value="ordinarySubscriber"/>
      <xs:enumeration value="subscriberWithPriority"/>
      <xs:enumeration value="dataCall"/>
      <xs:enumeration value="testCall"/>
      <xs:enumeration value="spare"/>
      <xs:enumeration value="payphone"/>
      <xs:enumeration value="category0"/>
      <xs:enumeration value="hotelsSubscriber"/>
      <xs:enumeration value="freeSubscriber"/>
      <xs:enumeration value="paidSubscriber"/>
      <xs:enumeration value="localSubscriber"/>
      <xs:enumeration value="localTaksofon"/>
      <xs:enumeration value="autoCallI"/>
      <xs:enumeration value="semiautoCallI"/>
      <xs:enumeration value="autoCallII"/>
      <xs:enumeration value="semiautoCallII"/>
      <xs:enumeration value="autoCallIII"/>
      <xs:enumeration value="semiautoCallIII"/>
      <xs:enumeration value="autoCallIV"/>
      <xs:enumeration value="semiautoCallIV"/>
      <xs:enumeration value="0"/>
      <xs:enumeration value="1"/>
      <xs:enumeration value="2"/>
      <xs:enumeration value="3"/>
      <xs:enumeration value="4"/>
      <xs:enumeration value="5"/>
      <xs:enumeration value="6"/>
      <xs:enumeration value="7"/>
      <xs:enumeration value="8"/>
      <xs:enumeration value="9"/>
    </xs:restriction>
  </xs:simpleType>
</xs:schema>

```

```
<xs:enumeration value="10"/>
<xs:enumeration value="11"/>
<xs:enumeration value="12"/>
<xs:enumeration value="13"/>
<xs:enumeration value="14"/>
<xs:enumeration value="15"/>
<xs:enumeration value="16"/>
<xs:enumeration value="17"/>
<xs:enumeration value="18"/>
<xs:enumeration value="19"/>
<xs:enumeration value="20"/>
<xs:enumeration value="21"/>
<xs:enumeration value="22"/>
<xs:enumeration value="23"/>
<xs:enumeration value="24"/>
<xs:enumeration value="25"/>
<xs:enumeration value="26"/>
<xs:enumeration value="27"/>
<xs:enumeration value="28"/>
<xs:enumeration value="29"/>
<xs:enumeration value="30"/>
<xs:enumeration value="31"/>
<xs:enumeration value="32"/>
<xs:enumeration value="33"/>
<xs:enumeration value="34"/>
<xs:enumeration value="35"/>
<xs:enumeration value="36"/>
<xs:enumeration value="37"/>
<xs:enumeration value="38"/>
<xs:enumeration value="39"/>
<xs:enumeration value="40"/>
<xs:enumeration value="41"/>
<xs:enumeration value="42"/>
<xs:enumeration value="43"/>
<xs:enumeration value="44"/>
<xs:enumeration value="45"/>
<xs:enumeration value="46"/>
<xs:enumeration value="47"/>
<xs:enumeration value="48"/>
<xs:enumeration value="49"/>
<xs:enumeration value="50"/>
<xs:enumeration value="51"/>
<xs:enumeration value="52"/>
<xs:enumeration value="53"/>
<xs:enumeration value="54"/>
<xs:enumeration value="55"/>
<xs:enumeration value="56"/>
<xs:enumeration value="57"/>
<xs:enumeration value="58"/>
<xs:enumeration value="59"/>
```

```
<xs:enumeration value="60"/>
<xs:enumeration value="61"/>
<xs:enumeration value="62"/>
<xs:enumeration value="63"/>
<xs:enumeration value="64"/>
<xs:enumeration value="65"/>
<xs:enumeration value="66"/>
<xs:enumeration value="67"/>
<xs:enumeration value="68"/>
<xs:enumeration value="69"/>
<xs:enumeration value="70"/>
<xs:enumeration value="71"/>
<xs:enumeration value="72"/>
<xs:enumeration value="73"/>
<xs:enumeration value="74"/>
<xs:enumeration value="75"/>
<xs:enumeration value="76"/>
<xs:enumeration value="77"/>
<xs:enumeration value="78"/>
<xs:enumeration value="79"/>
<xs:enumeration value="80"/>
<xs:enumeration value="81"/>
<xs:enumeration value="82"/>
<xs:enumeration value="83"/>
<xs:enumeration value="84"/>
<xs:enumeration value="85"/>
<xs:enumeration value="86"/>
<xs:enumeration value="87"/>
<xs:enumeration value="88"/>
<xs:enumeration value="89"/>
<xs:enumeration value="90"/>
<xs:enumeration value="91"/>
<xs:enumeration value="92"/>
<xs:enumeration value="93"/>
<xs:enumeration value="94"/>
<xs:enumeration value="95"/>
<xs:enumeration value="96"/>
<xs:enumeration value="97"/>
<xs:enumeration value="98"/>
<xs:enumeration value="99"/>
<xs:enumeration value="100"/>
<xs:enumeration value="101"/>
<xs:enumeration value="102"/>
<xs:enumeration value="103"/>
<xs:enumeration value="104"/>
<xs:enumeration value="105"/>
<xs:enumeration value="106"/>
<xs:enumeration value="107"/>
<xs:enumeration value="108"/>
<xs:enumeration value="109"/>
```

```
<xs:enumeration value="110"/>
<xs:enumeration value="111"/>
<xs:enumeration value="112"/>
<xs:enumeration value="113"/>
<xs:enumeration value="114"/>
<xs:enumeration value="115"/>
<xs:enumeration value="116"/>
<xs:enumeration value="117"/>
<xs:enumeration value="118"/>
<xs:enumeration value="119"/>
<xs:enumeration value="120"/>
<xs:enumeration value="121"/>
<xs:enumeration value="122"/>
<xs:enumeration value="123"/>
<xs:enumeration value="124"/>
<xs:enumeration value="125"/>
<xs:enumeration value="126"/>
<xs:enumeration value="127"/>
<xs:enumeration value="128"/>
<xs:enumeration value="129"/>
<xs:enumeration value="130"/>
<xs:enumeration value="131"/>
<xs:enumeration value="132"/>
<xs:enumeration value="133"/>
<xs:enumeration value="134"/>
<xs:enumeration value="135"/>
<xs:enumeration value="136"/>
<xs:enumeration value="137"/>
<xs:enumeration value="138"/>
<xs:enumeration value="139"/>
<xs:enumeration value="140"/>
<xs:enumeration value="141"/>
<xs:enumeration value="142"/>
<xs:enumeration value="143"/>
<xs:enumeration value="144"/>
<xs:enumeration value="145"/>
<xs:enumeration value="146"/>
<xs:enumeration value="147"/>
<xs:enumeration value="148"/>
<xs:enumeration value="149"/>
<xs:enumeration value="150"/>
<xs:enumeration value="151"/>
<xs:enumeration value="152"/>
<xs:enumeration value="153"/>
<xs:enumeration value="154"/>
<xs:enumeration value="155"/>
<xs:enumeration value="156"/>
<xs:enumeration value="157"/>
<xs:enumeration value="158"/>
<xs:enumeration value="159"/>
```



```
<xs:enumeration value="160"/>
<xs:enumeration value="161"/>
<xs:enumeration value="162"/>
<xs:enumeration value="163"/>
<xs:enumeration value="164"/>
<xs:enumeration value="165"/>
<xs:enumeration value="166"/>
<xs:enumeration value="167"/>
<xs:enumeration value="168"/>
<xs:enumeration value="169"/>
<xs:enumeration value="170"/>
<xs:enumeration value="171"/>
<xs:enumeration value="172"/>
<xs:enumeration value="173"/>
<xs:enumeration value="174"/>
<xs:enumeration value="175"/>
<xs:enumeration value="176"/>
<xs:enumeration value="177"/>
<xs:enumeration value="178"/>
<xs:enumeration value="179"/>
<xs:enumeration value="180"/>
<xs:enumeration value="181"/>
<xs:enumeration value="182"/>
<xs:enumeration value="183"/>
<xs:enumeration value="184"/>
<xs:enumeration value="185"/>
<xs:enumeration value="186"/>
<xs:enumeration value="187"/>
<xs:enumeration value="188"/>
<xs:enumeration value="189"/>
<xs:enumeration value="190"/>
<xs:enumeration value="191"/>
<xs:enumeration value="192"/>
<xs:enumeration value="193"/>
<xs:enumeration value="194"/>
<xs:enumeration value="195"/>
<xs:enumeration value="196"/>
<xs:enumeration value="197"/>
<xs:enumeration value="198"/>
<xs:enumeration value="199"/>
<xs:enumeration value="200"/>
<xs:enumeration value="201"/>
<xs:enumeration value="202"/>
<xs:enumeration value="203"/>
<xs:enumeration value="204"/>
<xs:enumeration value="205"/>
<xs:enumeration value="206"/>
<xs:enumeration value="207"/>
<xs:enumeration value="208"/>
<xs:enumeration value="209"/>
```

```
<xs:enumeration value="210"/>
<xs:enumeration value="211"/>
<xs:enumeration value="212"/>
<xs:enumeration value="213"/>
<xs:enumeration value="214"/>
<xs:enumeration value="215"/>
<xs:enumeration value="216"/>
<xs:enumeration value="217"/>
<xs:enumeration value="218"/>
<xs:enumeration value="219"/>
<xs:enumeration value="220"/>
<xs:enumeration value="221"/>
<xs:enumeration value="222"/>
<xs:enumeration value="223"/>
<xs:enumeration value="224"/>
<xs:enumeration value="225"/>
<xs:enumeration value="226"/>
<xs:enumeration value="227"/>
<xs:enumeration value="228"/>
<xs:enumeration value="229"/>
<xs:enumeration value="230"/>
<xs:enumeration value="231"/>
<xs:enumeration value="232"/>
<xs:enumeration value="233"/>
<xs:enumeration value="234"/>
<xs:enumeration value="235"/>
<xs:enumeration value="236"/>
<xs:enumeration value="237"/>
<xs:enumeration value="238"/>
<xs:enumeration value="239"/>
<xs:enumeration value="240"/>
<xs:enumeration value="241"/>
<xs:enumeration value="242"/>
<xs:enumeration value="243"/>
<xs:enumeration value="244"/>
<xs:enumeration value="245"/>
<xs:enumeration value="246"/>
<xs:enumeration value="247"/>
<xs:enumeration value="248"/>
<xs:enumeration value="249"/>
<xs:enumeration value="250"/>
<xs:enumeration value="251"/>
<xs:enumeration value="252"/>
<xs:enumeration value="253"/>
<xs:enumeration value="254"/>
<xs:enumeration value="255"/>
</xs:restriction>
</xs:simpleType>
<!-- naiType -->
<xs:simpleType name="naiType">
```

```

    <xs:restriction base="xs:string">
      <xs:enumeration value="spare"/>
      <xs:enumeration value="subscriberNumber"/>
      <xs:enumeration value="unknown"/>
      <xs:enumeration value="nationalNumber"/>
      <xs:enumeration value="internationalNumber"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- inniType -->
  <xs:simpleType name="inniType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="routingToInternalNumberAllowed"/>
      <xs:enumeration value="routingToInternalNumberNotAllowed"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- npiType -->
  <xs:simpleType name="npiType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="spare"/>
      <xs:enumeration value="isdnTelephony"/>
      <xs:enumeration value="dataNumberingPlan"/>
      <xs:enumeration value="telexNumberingPlan"/>
      <xs:enumeration value="reserved1"/>
      <xs:enumeration value="reserved2"/>
      <xs:enumeration value="reserved3"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- niType -->
  <xs:simpleType name="niType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="private"/>
      <xs:enumeration value="local"/>
      <xs:enumeration value="zone"/>
      <xs:enumeration value="intercity"/>
      <xs:enumeration value="international"/>
      <xs:enumeration value="emergency"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- apriType -->
  <xs:simpleType name="apriType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="presentationAllowed"/>
      <xs:enumeration value="presentationRestricted"/>
      <xs:enumeration value="addressNotAvailable"/>
      <xs:enumeration value="spare"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- screeningType -->
  <xs:simpleType name="screeningType">
    <xs:restriction base="xs:string">

```

```

        <xs:enumeration value="userProvidedNotVerified"/>
        <xs:enumeration value="userProvidedVerifiedAndPassed"/>
        <xs:enumeration value="userProvidedVerifiedAndFailed"/>
        <xs:enumeration value="networkProvided"/>
    </xs:restriction>
</xs:simpleType>
<!-- booleanType -->
<xs:simpleType name="booleanType">
    <xs:restriction base="xs:boolean"/>
</xs:simpleType>
<!-- ruleType -->
<xs:complexType name="ruleType">
    <xs:sequence>
        <xs:element name="conditions" minOccurs="0">
            <xs:complexType>
                <xs:all>
                    <xs:element name="cgpn" type="cgpnType" minOccurs="0"/>
                    <xs:element name="cdpn" type="cdpnType" minOccurs="0"/>
                    <xs:element name="rgn" type="rgnConditionType" minOccurs="0"/>
                    <xs:element name="rnn" type="rnnConditionType" minOccurs="0"/>
                    <xs:element name="ocdpn" type="ocdpnConditionType" minOccurs="0"/>
                    <xs:element name="cn" type="cnConditionType" minOccurs="0"/>
                    <xs:element name="time" type="valueType" minOccurs="0"/>
                    <xs:element name="date" type="valueType" minOccurs="0"/>
                    <xs:element name="weekday" type="valueType" minOccurs="0"/>
                    <xs:element name="tag" type="valueType" minOccurs="0"/>
                </xs:all>
            </xs:complexType>
        </xs:element>
        <xs:element name="actions" minOccurs="0">
            <xs:complexType>
                <xs:all>
                    <xs:element name="cgpn" type="cgpnType" minOccurs="0"/>
                    <xs:element name="cdpn" type="cdpnType" minOccurs="0"/>
                    <xs:element name="rgn" type="rgnActionType" minOccurs="0"/>
                    <xs:element name="rnn" type="rnnActionType" minOccurs="0"/>
                    <xs:element name="ocdpn" type="ocdpnActionType" minOccurs="0"/>
                    <xs:element name="cn" type="cnActionType" minOccurs="0"/>
                    <xs:element name="empty_rgn" type="nullType" minOccurs="0">
                        <xs:annotation><xs:documentation xml:lang="en">Remove routing
number.</xs:documentation></xs:annotation>
                    </xs:element>
                    <xs:element name="empty_ocdpn" type="nullType" minOccurs="0">
                        <xs:annotation><xs:documentation xml:lang="en">Remove original called
number.</xs:documentation></xs:annotation>
                    </xs:element>
                    <xs:element name="empty_cn" type="nullType" minOccurs="0">
                        <xs:annotation><xs:documentation xml:lang="en">Remove connected
number.</xs:documentation></xs:annotation>
                    </xs:element>
                </xs:all>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>

```

```

        </xs:all>
    </xs:complexType>
</xs:element>
<xs:element name="result">
    <xs:complexType>
        <xs:choice>
            <xs:element name="finish" type="finishResultType"/>
            <xs:element name="error" type="errorResultType"/>
            <xs:element name="next" type="nextResultType"/>
            <xs:element name="continue" type="continueResultType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="name" type="xs:string" use="required"/>
<xs:attribute name="description" type="xs:string" use="optional"/>
</xs:complexType>
<!-- modifiersType -->
<xs:complexType name="modifiersType">
    <xs:all>
        <xs:element name="in" minOccurs="0" maxOccurs="1">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="rule" type="ruleType" maxOccurs="unbounded"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="out" minOccurs="0" maxOccurs="1">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="rule" type="ruleType" maxOccurs="unbounded"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:all>
    <xs:attribute name="name" type="xs:string" use="required">
        <xs:annotation>
            <xs:documentation>The name of the current modifiers.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="description" type="xs:string" use="optional">
        <xs:annotation>
            <xs:documentation>Text description of the current modifiers.</xs:documentation
>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>
<!-- adaptationType -->
<xs:complexType name="adaptationType">
    <xs:sequence>

```

```

        <xs:element name="rule" type="ruleType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required">
        <xs:annotation>
            <xs:documentation>The name of the current adaptation.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="description" type="xs:string" use="optional">
        <xs:annotation>
            <xs:documentation>Text description of the current adaptation.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>
<!-- modifiers -->
<xs:element name="modifiers" type="modifiersType"/>
<!-- adaptation -->
<xs:element name="adaptation" type="adaptationType"/>
<!-- cdpnType -->
<xs:complexType name="cdpnType">
    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="nai" type="naiType"/>
    <xs:attribute name="incomplete" type="xs:boolean"/>
    <xs:attribute name="inni" type="inniType"/>
    <xs:attribute name="npi" type="npiType"/>
    <xs:attribute name="ni" type="niType"/>
    <xs:attribute name="category" type="categoryType"/>
    <xs:attribute name="in_list" type="xs:string"/>
    <xs:attribute name="display_name" type="xs:string"/>
</xs:complexType>
<!-- cgpnType -->
<xs:complexType name="cgpnType">
    <xs:attribute name="nai" type="naiType"/>
    <xs:attribute name="incomplete" type="xs:boolean"/>
    <xs:attribute name="npi" type="npiType"/>
    <xs:attribute name="apri" type="apriType"/>
    <xs:attribute name="screening" type="screeningType"/>
    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="ni" type="niType"/>
    <xs:attribute name="in_list" type="xs:string"/>
    <xs:attribute name="display_name" type="xs:string"/>
    <xs:attribute name="caller_id" type="xs:string"/>
</xs:complexType>
<!-- rgnConditionType -->
<xs:complexType name="rgnConditionType">
    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="nai" type="naiType"/>
    <xs:attribute name="incomplete" type="xs:boolean"/>
    <xs:attribute name="apri" type="apriType"/>
    <xs:attribute name="npi" type="npiType"/>
    <xs:attribute name="ni" type="niType"/>

```

```

    <xs:attribute name="in_list" type="xs:string"/>
    <xs:attribute name="empty" type="booleanType">
      <xs:annotation>
        <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
<!-- rnnConditionType -->
<xs:complexType name="rnnConditionType">
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
  <xs:attribute name="inni" type="inniType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="ni" type="niType"/>
  <xs:attribute name="in_list" type="xs:string"/>
  <xs:attribute name="empty" type="booleanType">
    <xs:annotation>
      <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
<!-- ocdpnConditionType -->
<xs:complexType name="ocdpnConditionType">
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="ni" type="niType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="apri" type="apriType"/>
  <xs:attribute name="category" type="categoryType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
  <xs:attribute name="in_list" type="xs:string"/>
  <xs:attribute name="empty" type="booleanType">
    <xs:annotation>
      <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
<!-- cnConditionType -->
<xs:complexType name="cnConditionType">
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="apri" type="apriType"/>
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="ni" type="niType"/>
  <xs:attribute name="in_list" type="xs:string"/>
  <xs:attribute name="empty" type="booleanType">

```

```

    <xs:annotation>
      <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
<!-- rgnActionType -->
<xs:complexType name="rgnActionType">
  <xs:attribute name="display_name" type="xs:string"/>
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
  <xs:attribute name="apri" type="apriType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="ni" type="niType"/>
</xs:complexType>
<!-- rnnActionType -->
<xs:complexType name="rnnActionType">
  <xs:attribute name="display_name" type="xs:string"/>
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
  <xs:attribute name="inni" type="inniType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="ni" type="niType"/>
</xs:complexType>
<!-- ocdpnActionType -->
<xs:complexType name="ocdpnActionType">
  <xs:attribute name="display_name" type="xs:string"/>
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="ni" type="niType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="apri" type="apriType"/>
  <xs:attribute name="category" type="categoryType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
</xs:complexType>
<!-- cnActionType -->
<xs:complexType name="cnActionType">
  <xs:attribute name="display_name" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="apri" type="apriType"/>
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="ni" type="niType"/>
  <xs:attribute name="caller_id" type="xs:string"/>
  <xs:attribute name="empty" type="booleanType">
    <xs:annotation>
      <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
  
```



```

        </xs:annotation>
    </xs:attribute>
</xs:complexType>
<!-- valueType -->
<xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:string" use="required"/>
</xs:complexType>
<!-- nullType -->
<xs:complexType name="nullType"/>
<!-- Result types -->
<!-- finishResultType -->
<xs:complexType name="finishResultType"/>
<!-- errorResultType -->
<xs:complexType name="errorResultType">
    <xs:attribute name="acp_cause" type="xs:string" use="optional"/>
    <xs:attribute name="isup_cause" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
</xs:complexType>
<!-- continueResultTypeType -->
<xs:simpleType name="continueResultTypeType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="start"/>
        <xs:enumeration value="next"/>
    </xs:restriction>
</xs:simpleType>
<!-- nextResultType -->
<xs:complexType name="nextResultType">
    <xs:attribute name="tag" type="xs:string" use="optional"/>
</xs:complexType>
<!-- continueResultType -->
<xs:complexType name="continueResultType">
    <xs:attribute name="tag" type="xs:string" use="optional"/>
    <xs:attribute name="type" type="continueResultTypeType" default="start" use="optional"/>
</xs:complexType>
</xs:schema>

```

Пример 1:

Запрос существующего контекста

Запрос:

http://192.168.1.21:9999/commands/adaptation_context_show

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_adaptation_context_show.xsd">
    <request domain="biysk.local" context="adp_sorm"/>
</in>

```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_adaptat
  ion_context_show.xsd">
<context
>PD94bWwgdMvyc2lvcj0iMS4wIiB1bmNvZGluZz0iVVRGLTgiPz48YWRhcHRhdGlvbiB4bWxuczp4cz0iaHR0cDovL3d3dy53
My5vcmcvMjAwMSYtUxTY2hlbWEtaW5zdGFuY2UiIHh0M5vTmFtZXNwYWNlU2NoZW1hTG9jYXRpb249ImVjc3NfbW9kawZpY
2F0b3JzLnhzZCIgYmFtZT0iYWRwX3NvcmlPjxydWx1IG5hbWU9ImZvc19zb3JtIj48Y29uZG10aw9ucy8+PGFjdGlvbnMvPj
xyZXN1bHQ+PGZpbmlzaC8+PC9yZXN1bHQ+PC9ydWx1PjwvYWRhcHRhdGlvbj4=</context>
</out>
```

Пример 2:

Запрос несуществующего контекста

Запрос:

http://192.168.1.21:9999/commands/adaptation_context_show

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_adaptation_context_show.xsd">
  <request domain="biysk.local" context="ctx1"/>
</in>
```

Ответ:

404

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_adaptat
  ion_context_show.xsd">
  <error cmd="adaptation_context_show" reason="context_not_found" body="{error,{ctx_not_found,
  {"biysk.local",ctx1}}}" entity="ctx1"/>
</out>
```

Hc_address_book_options - Управление настройками подключения к БД "Адресная книга"

- [Hc_address_book_options_clean](#)
- [Hc_address_book_options_info](#)
- [Hc_address_book_options_set](#)

`hc_address_book_options_clean`

Команда сброса настроек подключения к БД "Адресная книга".

Аналог команды в CoCon:

`system/address-book/properties/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/address_book_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemCleanOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/address_book_options_clean

```

<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="address_book_options_clean.xsd">
  <request>
    <system_options>
      <system>
        <option name="username"/>
        <option name="port"/>
        <option name="password"/>
        <option name="host"/>
        <option name="database"/>
        <option name="auto_delete_account"/>
        <option name="auto_create_account"/>
      </system>
    </system_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="address_book_options_clean.xsd">
  <result>
    <system_options>
      <system>
        <success option="auto_create_account"/>
        <success option="auto_delete_account"/>
        <success option="database"/>
        <success option="host"/>
        <success option="password"/>
        <success option="port"/>
        <success option="username"/>
      </system>
    </system_options>
  </result>
</out>

```

[Hc_address_book_options_info](#)

Команда просмотра настроек подключения к БД "Адресная книга".

Аналог команды в CoCon:

system/address-book/properties/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/address_book_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- addressBookOptionsType -->
  <xs:complexType name="addressBookOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="auto_delete_account" type="booleanType"/>
        <xs:attribute name="auto_create_account" type="booleanType"/>
        <xs:attribute name="database" type="stringType"/>
        <xs:attribute name="host" type="stringType"/>
        <xs:attribute name="password" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              In case of strong security restrictions password may be shown as
*****
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="port" type="inetPortType"/>
        <xs:attribute name="username" type="stringType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemInfoInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>
            <xs:all>

```

```

        <xs:element name="system_options" type="systemInfoOutType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/address_book_options_info

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="address_book_options_info.xsd">
  <request>
    <system_options/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="address_book_options_info.xsd">
  <result>
    <system_options>
      <system>
        <options xs:type="addressBookOptionsType" auto_create_account="true"
          auto_delete_account="true" database="ecss_address_book" host="address-book.mysql.ecss"
          password="*****" port="3306" username="address_book"/>
      </system>
    </system_options>
  </result>
</out>

```

hc_address_book_options_set

Команда изменения настроек настроек подключения к БД "Адресная книга".

Аналог команды в CoCon:

system/address-book/properties/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/address_book_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- addressBookOptionsType -->
  <xs:complexType name="addressBookOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="auto_delete_account" type="booleanType"/>
        <xs:attribute name="auto_create_account" type="booleanType"/>
        <xs:attribute name="database" type="stringType"/>
        <xs:attribute name="host" type="stringType"/>
        <xs:attribute name="password" type="stringType"/>
        <xs:attribute name="port" type="inetPortType"/>
        <xs:attribute name="username" type="stringType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemSetOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```
</xs:element>
</xs:schema>
```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/address_book_options_set

```
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="address_book_options_set.xsd">
  <request>
    <system_options>
      <system>
        <options auto_create_account="true" auto_delete_account="true"
database="ecss_address_book" host="address-book.mysql.ecss" password="address_book" port="3306"
username="address_book" xs:type="addressBookOptionsType"/>
      </system>
    </system_options>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="address_book_options_set.xsd">
  <result>
    <system_options>
      <system>
        <success option="username"/>
        <success option="port"/>
        <success option="password"/>
        <success option="host"/>
        <success option="database"/>
        <success option="auto_delete_account"/>
        <success option="auto_create_account"/>
      </system>
    </system_options>
  </result>
</out>
```

Нс_agent - Управление агентами виртуального call-центра

- [Hc_agent_clean](#)
- [Hc_agent_declare](#)
- [Hc_agent_info](#)
- [Hc_agent_list](#)
- [Hc_agent_remove](#)
- [Hc_agent_set](#)

Hc_agent_clean

Команда позволяет очистить свойства определенных агентов виртуального call-центра.

Аналог команды в CoSop:

`/domain/<DOMAIN>/cc/agent/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/agent_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>

  <!--Property name-->
  <xs:complexType name="propertyNameType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!--warningReasonType-->
  <xs:simpleType name="warningReasonType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="not_exists"/>
    </xs:restriction>
  </xs:simpleType>
  <!--warningsType-->
  <xs:complexType name="warningsType">
    <xs:sequence>
      <xs:element name="warning" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="agent" type="xs:string"/>
          <xs:attribute name="reason" type="warningReasonType"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <!--propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!--cleanAgentType-->
  <xs:complexType name="cleanAgentType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!--agentsType-->
  <xs:complexType name="agentsType">
    <xs:sequence>
      <xs:element name="agent" type="cleanAgentType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">

```

```

<xs:complexType>
  <xs:sequence>
    <!--request body-->
    <xs:element name="agents" type="agentsType"/>
    <!--end-->
  </xs:sequence>
  <!--request attributes-->
  <xs:attribute name="domain" use="required"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!--out-->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="response">
        <xs:complexType>
          <xs:choice>
            <!--all ok-->
            <xs:element name="ok"/>
            <!--not very bad. command works fine but some agents not exists-->
            <xs:element name="warnings" type="warningsType"/>
          </xs:choice>
        </xs:complexType>
      </xs:element>
      <!--bad. command failed-->
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1

Запрос:

http://192.168.1.21:9999/commands/agent_clean

```
<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request domain="biysk.local">
    <agents>
      <agent name="516">
        <property name="description"/>
        <property name="custom\key\unicode2"/>
      </agent>
      <agent name="515"/>
    </agents>
  </request>
</in>
```

Ответ 200:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_agent_
clean.xsd">
  <response>
    <ok/>
  </response>
</out>
```

Пример 2

Ответ 200 на предыдущий запрос если часть агентов не существует:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_agent_c
lean.xsd">
  <response>
    <warnings>
      <warning agent="555" reason="not_exists"/>
    </warnings>
  </response>
</out>
```

Пример 3

Ответ 434 на тот же самый запрос, если нет такого домена:

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_agent_lean.xsd">
  <error cmd="agent_clean" reason="nocommand" body="Command not found by path /domain/d.test/cc/agent/clean"/>
</out>
```

Нс_agent_declare

Команда позволяет создать агентов виртуального call-центра.

Аналог команды в CoCon:

`domain/<DOMAIN>/cc/agent/declare`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/agent_declare`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="hc_agent_common.xsd"/>
  <xs:include schemaLocation="error.xsd"/>
  <!--agentsType-->
  <xs:complexType name="agentsType">
    <xs:sequence>
      <xs:element name="agent" type="agentConfigurationType" minOccurs="0" maxOccurs="unbounded"/
>
    </xs:sequence>
  </xs:complexType>
  <!--warningReasonType-->
  <xs:simpleType name="warningReasonType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="already_declared"/>
    </xs:restriction>
  </xs:simpleType>
  <!--warningsType-->
  <xs:complexType name="warningsType">
    <xs:sequence>
      <xs:element name="warning" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="agent" type="xs:string"/>
          <xs:attribute name="reason" type="warningReasonType"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <!--in-->
  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <!--request body-->
              <xs:element name="agents" type="agentsType"/>
            </xs:sequence>
            <xs:attribute name="domain" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name = "out">
    <xs:complexType>

```

```
<xs:choice>
  <xs:element name="response">
    <xs:complexType>
      <xs:choice>
        <!--all agents was declared-->
        <xs:element name="ok"/>
        <!--not very bad. command works fine but some agents are already declared-->
        <xs:element name="warnings" type="warningsType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!--bad. command failed:-->
  <xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
```

XSD-схема hc_agent_common.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="common_types.xsd"/>

  <!--autoCompleteTimeoutType-->
  <xs:simpleType name="autoCompleteTimeoutType">
    <xs:restriction base="xs:nonNegativeInteger">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="3600"/>
    </xs:restriction>
  </xs:simpleType>

  <!--autoLogoutTimeoutType-->
  <xs:simpleType name="autoLogoutTimeoutType">
    <xs:restriction base="xs:nonNegativeInteger">
      <xs:minInclusive value="5"/>
      <xs:maxInclusive value="720"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="workingTimeoutType">
    <xs:restriction base="xs:nonNegativeInteger">
      <xs:minInclusive value="1"/>
      <xs:maxInclusive value="1440"/>
    </xs:restriction>
  </xs:simpleType>

  <!--Skills-->
  <xs:complexType name="skillType">
    <xs:attribute name="type" type="xs:string" use="required"/>
    <xs:attribute name="subtype" type="xs:string" use="optional"/>
    <xs:attribute name="level" type="xs:integer" use="required"/>
  </xs:complexType>
  <xs:complexType name="skillsType">
    <xs:sequence>
      <xs:element name="skill" type="skillType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!--Custom properties-->
  <xs:complexType name="customPropertyType">
    <xs:attribute name="name" type="xs:string"/>
    <xs:attribute name="value" type="xs:string"/>
  </xs:complexType>
  <xs:complexType name="customPropertiesType">
    <xs:sequence>
      <xs:element name="property" type="customPropertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

```

```

</xs:complexType>
<!--Agent configuration-->
<xs:complexType name="agentConfigurationType">
  <xs:all>
    <xs:element name="skills" type="skillsType" minOccurs="0"/>
    <xs:element name="properties" type="customPropertiesType" minOccurs="0"/>
  </xs:all>
  <xs:attribute name="id" type="xs:string" use="required"/>
  <xs:attribute name="password" type="xs:string" use="optional"/>
  <xs:attribute name="group" type="xs:string" use="optional"/>
  <xs:attribute name="description" type="xs:string" use="optional"/>
  <xs:attribute name="display_name" type="xs:string" use="optional"/>
  <xs:attribute name="supervisor" type="xs:boolean" use="optional"/>
  <xs:attribute name="auto_complete" type="xs:boolean" use="optional"/>
  <xs:attribute name="auto_complete_timeout" type="autoCompleteTimeoutType" use="optional"/>
  <xs:attribute name="call_recording" type="xs:boolean" use="optional"/>
  <xs:attribute name="pickup" type="xs:boolean" use="optional"/>
  <xs:attribute name="load" type="xs:positiveInteger" use="optional"/>
  <xs:attribute name="working_timeout" type="workingTimeoutType" use="optional"/>
  <xs:attribute name="auto_logout_timeout" type="autoLogoutTimeoutType" use="optional"/>
</xs:complexType>
<!--Agent-->
<xs:complexType name="agentType">
  <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>
</xs:schema>

```

Пример 1

Запрос:

http://192.168.1.21:9999/commands/agent_declare

```

<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request domain="biysk.local">
    <agents>
      <agent id="ag010" password="1111" group="group1" auto_complete="false" load="1"
description="" supervisor="true" display_name="Фёдор" call_recording="true">
        <skills>
          <skill type="language" subtype="english" level="14234"/>
          <skill type="fire" level="2"/>
        </skills>
        <properties>
          <property name="custom\key\term" value="{value}"/>
          <property name="custom\key\unicode" value="&quot;Значение1&quot;"/>
        </properties>
      </agent>
      <agent id="ag020" password="1234" group="group2" auto_complete="false" load="1"
description="" supervisor="true" display_name="Vasya" call_recording="true">
        <skills>
          <skill type="language" subtype="russian" level="2"/>
          <skill type="fire" level="21"/>
        </skills>
      </agent>
    </agents>
  </request>
</in>

```

Ответ 200:

```

<?xml version="1.0" encoding="UTF-8"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_agent_
declare.xsd">
  <response>
    <ok />
  </response>
</out>

```

Пример 2

Ответ на предыдущий запрос если некоторые агенты уже существуют:

```
<?xml version="1.0" encoding="UTF-8"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_agent_
declare.xsd">
  <response>
    <warnings>
      <warning agent="ag010" reason="already_declared" />
      <warning agent="ag020" reason="already_declared" />
    </warnings>
  </response>
</out>
```

Пример 3

Ответ 434 на предыдущий запрос, если нет такого домена:

```
<?xml version="1.0" encoding="UTF-8"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_agent_
declare.xsd">
  <error cmd="agent_declare" reason="nocommand" body="Command not found by path /domain/d.test/
cc/agent/declare" />
</out>
```

hc_agent_info

Команда позволяет вывести информацию об агентах виртуального call-центра в режиме реального времени.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/agent/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/agent_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="hc_agent_common.xsd"/>
  <xs:include schemaLocation="error.xsd"/>
  <!--agentsType-->
  <xs:complexType name="agentsType">
    <xs:sequence>
      <xs:element name="agent" type="agentType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!--in-->
  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <!--request body-->
              <xs:element name="agents" type="agentsType"/>
              <!--end-->
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name = "out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:all>
              <!-- response body -->
              <xs:element name="agents" minOccurs="0" maxOccurs="1">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="agent" type="agentConfigurationType" minOccurs="0"
maxOccurs="unbounded"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
              <xs:element name="warnings" minOccurs="0" maxOccurs="1">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="warning" minOccurs="0" maxOccurs="unbounded">

```

```

        <xs:complexType>
          <xs:attribute name="agent" type="xs:string"/>
          <xs:attribute name="reason" type="xs:string"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- response end -->
</xs:all>
</xs:complexType>
</xs:element>
<!--Or error-->
<xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/agent_info

```

<?xml version="1.0" encoding="utf-8"?>
<in>
  <request domain="biysk.local">
    <agents>
      <agent name="500"/>
      <agent name="502"/>
      <agent name="514"/>
      <agent name="517"/>
    </agents>
  </request>
</in>

```

Ответ:


```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_agent_i
nfo.xsd">
  <response>
    <agents>
      <agent id="500" password="500500" group="ltp1" auto_complete="true" load="1"
description="" supervisor="true" display_name="" call_recording="false" pickup="false"
auto_complete_timeout="5" working_timeout="120">
        <properties>
          <property name="occupied_phone_number" value="240101"/>
        </properties>
      </agent>
      <agent id="502" password="500500" group="ltp1" auto_complete="true" load="1"
description="" supervisor="false" display_name="" call_recording="false" pickup="false"
auto_complete_timeout="5" working_timeout="120"/>
      <agent id="514" password="500500" group="ltp2" auto_complete="true" load="1"
description="" supervisor="false" display_name="" call_recording="false" pickup="false"
auto_complete_timeout="5" working_timeout="120"/>
      <agent id="517" password="500500" group="ltp2" auto_complete="true" load="1"
description="" supervisor="false" display_name="" call_recording="false" pickup="false"
auto_complete_timeout="5" working_timeout="120"/>
    </agents>
  </response>
</out>

```

hc_agent_list

Команда позволяет вывести список агентов виртуального call-центра.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/agent/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/agent_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="hc_agent_common.xsd"/>

  <!--agentsType-->
  <xs:complexType name="agentsType">
    <xs:sequence>
      <xs:element name="agent" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="id" type="xs:nonNegativeInteger" use="required"/>
          <xs:attribute name="group" type="xs:string" use="optional"/>
          <xs:attribute name="display_name" type="xs:string" use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name = "out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name = "agents" type="agentsType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/agent_list

```
<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request domain="biysk.local"/>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_agent_1
ist.xsd">
  <response>
    <agents>
      <agent id="504" display_name="" group="ltp1"/>
      <agent id="509" display_name="" group="ltp1"/>
      <agent id="502" display_name="" group="ltp1"/>
      <agent id="506" display_name="" group="ltp1"/>
      <agent id="518" display_name="" group="ltp2"/>
      <agent id="523" display_name="" group="ltp3"/>
      <agent id="501" display_name="" group="ltp1"/>
      <agent id="508" display_name="" group="ltp1"/>
      <agent id="521" display_name="" group="ltp3"/>
      <agent id="510" display_name="" group="ltp2"/>
      <agent id="516" display_name="" group="ltp2"/>
      <agent id="527" display_name="" group="ltp3"/>
      <agent id="514" display_name="" group="ltp2"/>
      <agent id="525" display_name="" group="ltp3"/>
      <agent id="520" display_name="" group="ltp3"/>
      <agent id="513" display_name="" group="ltp2"/>
      <agent id="507" display_name="" group="ltp1"/>
      <agent id="512" display_name="" group="ltp2"/>
      <agent id="517" display_name="" group="ltp2"/>
      <agent id="529" display_name="" group="ltp3"/>
      <agent id="515" display_name="" group="ltp2"/>
      <agent id="528" display_name="" group="ltp3"/>
      <agent id="519" display_name="" group="ltp2"/>
      <agent id="505" display_name="" group="ltp1"/>
      <agent id="526" display_name="" group="ltp3"/>
      <agent id="503" display_name="" group="ltp1"/>
      <agent id="500" display_name="" group="ltp1"/>
      <agent id="511" display_name="" group="ltp2"/>
      <agent id="524" display_name="" group="ltp3"/>
      <agent id="522" display_name="" group="ltp3"/>
    </agents>
  </response>
</out>
```

Нс_agent_remove

Команда позволяет удалить указанных агентов виртуального call-центра.

Аналог команды в CoCon:

`domain/<DOMAIN>/cc/agent/remove`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/agent_remove

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="hc_agent_common.xsd"/>

  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <!--request body-->
              <xs:element name="agents">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="agent" type="agentType" maxOccurs="unbounded"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
              <!--end-->
            </xs:sequence>
            <xs:attribute name="domain" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name = "out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/agent_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request domain="biysk.local">
    <agents>
      <agent name="519"/>
      <agent name="520"/>
    </agents>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_agent_remove.xsd">
  <response>
    <ok/>
  </response>
</out>
```

Нс_agent_set

Команда позволяет изменить свойства определенных агентов виртуального call-центра.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/agent/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/agent_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="hc_agent_common.xsd"/>
  <xs:include schemaLocation="error.xsd"/>
  <!--agentsType-->
  <xs:complexType name="agentsType">
    <xs:sequence>
      <xs:element name="agent" type="agentConfigurationType" minOccurs="0" maxOccurs="unbounded"/
>
    </xs:sequence>
  </xs:complexType>
  <!--requestType-->
  <xs:complexType name="requestType">
    <xs:sequence>
      <!--request body-->
      <xs:element name="agents" type="agentsType"/>
    </xs:sequence>
    <xs:attribute name="domain" use="required"/>
  </xs:complexType>
  <!--responseType-->
  <xs:complexType name="responseType">
    <xs:choice>
      <!--all ok-->
      <xs:element name="ok"/>
      <!--not very bad. command works fine but some agents not exists-->
      <xs:element name="warnings" type="warningsType"/>
    </xs:choice>
  </xs:complexType>
  <!--warningReasonType-->
  <xs:simpleType name="warningReasonType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="not_exists"/>
      <xs:enumeration value="badarg"/>
      <xs:enumeration value="unknown_error"/>
    </xs:restriction>
  </xs:simpleType>
  <!--warningsType-->
  <xs:complexType name="warningsType">
    <xs:sequence>
      <xs:element name="warning" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="agent" type="xs:string" use="required"/><?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_agent_1
ist.xsd">
  <response>
    <agents>

```

```

<agent id="504" display_name="" group="ltp1"/>
<agent id="509" display_name="" group="ltp1"/>
<agent id="502" display_name="" group="ltp1"/>
<agent id="506" display_name="" group="ltp1"/>
<agent id="518" display_name="" group="ltp2"/>
<agent id="523" display_name="" group="ltp3"/>
<agent id="501" display_name="" group="ltp1"/>
<agent id="508" display_name="" group="ltp1"/>
<agent id="510" display_name="" group="ltp2"/>
<agent id="521" display_name="" group="ltp3"/>
<agent id="516" display_name="" group="ltp2"/>
<agent id="514" display_name="" group="ltp2"/>
<agent id="527" display_name="" group="ltp3"/>
<agent id="525" display_name="" group="ltp3"/>
<agent id="ag020" display_name="Vasya" group="group2"/>
<agent id="520" display_name="" group="ltp3"/>
<agent id="513" display_name="" group="ltp2"/>
<agent id="507" display_name="" group="ltp1"/>
<agent id="517" display_name="" group="ltp2"/>
<agent id="512" display_name="" group="ltp2"/>
<agent id="529" display_name="" group="ltp3"/>
<agent id="515" display_name="" group="ltp2"/>
<agent id="528" display_name="" group="ltp3"/>
<agent id="519" display_name="" group="ltp2"/>
<agent id="505" display_name="" group="ltp1"/>
<agent id="ag010" display_name="Фёдор" group="group1"/>
<agent id="526" display_name="" group="ltp3"/>
<agent id="500" display_name="" group="ltp1"/>
<agent id="503" display_name="" group="ltp1"/>
<agent id="511" display_name="" group="ltp2"/>
<agent id="524" display_name="" group="ltp3"/>
<agent id="522" display_name="" group="ltp3"/>
</agents>
</response>
</out>
  <xs:attribute name="reason" type="warningReasonType" use="required"/>
  <xs:attribute name="body" type="xs:string" use="optional"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<!--in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request" type="requestType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!--out-->

```



```

<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="response" type="responseType"/>
      <!--bad. command failed!-->
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/agent_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request domain="biysk.local">
    <agents>
      <agent id="515" password="4321" group="group2" auto_complete="true" load="2"
description="New description" supervisor="false" display_name="Фёдор I" call_recording="false">
        <skills>
          <skill type="language" subtype="english" level="1"/>
        </skills>
        <properties>
          <property name="custom\key\term" value="{value, 1}"/>
          <property name="custom\key\unicode2" value="&quot;Значение 2&quot;"/>
        </properties>
      </agent>
      <agent id="516" password="1122"/>
    </agents>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_agent_s
et.xsd">
  <response>
    <ok/>
  </response>
</out>

```

Hc_alarms - Управление системой мониторинга и вывода аварий

- Hc_ap_speaker_off
- Hc_ap_status
- Hc_alarms_list
- Hc_alarms_mask_list
- Hc_node_rps_alarms
- Hc_alarms_mask_add
- Hc_alarms_mask_del
- Hc_alarms_clear
- Hc_alarms_delete
- Hc_alarm_notifiers_options_clean
- Hc_alarm_notifiers_options_info
- Hc_alarm_notifiers_options_set
- Hc_alarms_notifiers_send_test_msg

Hc_ap_speaker_off

Команда выключает динамик на блоке аварийной сигнализации (БАС).

Аналог команды в CoCon:

```
cluster/mediator/<CLUSTER>/ap/speaker-off
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ap_speaker_off

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 202 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="mediator" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос:

http://192.168.23.34:9999/commands/ap_speaker_off

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ap_speaker_off.xsd">
  <request mediator="md1"/>
</in>
```

Ответ:

202

Нс_ap_status

Команда возвращает состояние блока аварийной сигнализации (БАС).

Аналог команды в CoCon:

cluster/mediator/<CLUSTER>/ap/status

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:http://host:port/commands/ap_status**Код HTTP-ответа:**

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- errorType-->
  <xs:complexType name="errorType">
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- basType-->
  <xs:complexType name="basType">
    <xs:attribute name="ip" type="xs:string" use="required"/>
    <xs:attribute name="port" type="xs:integer" use="required"/>
    <xs:attribute name="speaker" type="xs:boolean" use="required"/>
    <xs:attribute name="led_0" type="xs:boolean" use="required"/>
    <xs:attribute name="led_1" type="xs:boolean" use="required"/>
    <xs:attribute name="led_2" type="xs:boolean" use="required"/>
    <xs:attribute name="led_3" type="xs:boolean" use="required"/>
    <xs:attribute name="led_4" type="xs:boolean" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="mediator" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType" minOccurs="0" />
        <xs:element name="bas" type="basType" minOccurs="0" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/ap_status

```
<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ap_status.xsd">
  <request mediator="md1"/>
</in>
```

Ответ:

200

```
<?xml version="1.0"?><out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ap_status.xsd"><bas ip="192.168.1.10" port="2222" speaker="false"
  led_0="true" led_1="false" led_2="false" led_3="false" led_4="false"/></out>
```

В случае ошибки:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ap_status.xsd">
  <error reason="Not connection to the alarm panel."/>
</out>
```

[Hc_alarms_list](#)

Команда возвращает список аварий в системе.

Аналог команды в CoCon:

```
cluster/mediator/<CLUSTER>/alarms/list
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alarms_list

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="rps.xsd"/>
  <!-- stringType -->
  <xs:complexType name="stringType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- pstrArgType -->
  <xs:complexType name="pstrArgType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- pstrType -->
  <xs:complexType name="pstrType">
    <xs:sequence minOccurs="0" maxOccurs="unbounded">
      <xs:element name="arg" type="pstrArgType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="value" type="xs:string" use="required"/>
    <xs:attribute name="id" type="xs:string" use="optional"/>
    <xs:attribute name="template" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:simpleType name="requestType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="active"/>
      <xs:enumeration value="cleared"/>
      <xs:enumeration value="all"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- operType -->
  <xs:simpleType name="operType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="equal"/>
      <xs:enumeration value="not_equal"/>
      <xs:enumeration value="more"/>
      <xs:enumeration value="less"/>
      <xs:enumeration value="more_or_equal"/>
      <xs:enumeration value="less_or_equal"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- sortType -->
  <xs:simpleType name="sortType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="asc"/>
      <xs:enumeration value="desc"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- columnType -->
  <xs:complexType name="columnType">

```

```

    <xs:attribute name="name" type="columnNameType" use="required"/>
    <xs:attribute name="sort" type="sortType" use="optional"/>
</xs:complexType>
<!-- filterType -->
<xs:complexType name="filterType">
  <xs:choice>
    <xs:element name="id" type="stringType"/>
    <xs:element name="managedObjectClass" type="stringType"/>
    <xs:element name="managedObjectInstance" type="stringType"/>
    <xs:element name="location" type="stringType"/>
    <xs:element name="domain" type="stringType"/>
    <xs:element name="user_id" type="stringType"/>
    <xs:element name="eventTime" type="eventTimeType"/>
    <!-- datetime or undefined -->
    <xs:element name="cleared" type="eventTimeType"/>
    <xs:element name="probableCause">
      <xs:complexType>
        <xs:attribute name="value" type="alarmCauseType" use="required"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="eventType">
      <xs:complexType>
        <xs:attribute name="value" type="alarmEventTypeType" use="required"/>
      </xs:complexType>
    </xs:element>
    <!-- term -->
    <xs:element name="specificProblems" type="stringType"/>
    <xs:element name="perceivedSeverity">
      <xs:complexType>
        <xs:attribute name="value" type="severityType" use="required"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="originalSeverity">
      <xs:complexType>
        <xs:attribute name="value" type="severityType" use="required"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="backedupStatus">
      <xs:complexType>
        <xs:attribute name="value" type="backedupStatusType" use="required"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="backupObject" type="stringType"/>
    <xs:element name="trendIndication">
      <xs:complexType>
        <xs:attribute name="value" type="trendIndicationType" use="required"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="thresholdInformation" type="stringType"/>
    <xs:element name="notificationIdentifier" type="stringType"/>

```

```

    <xs:element name="correlatedNotifications" type="stringType"/>
    <xs:element name="stateChangeDefinition" type="stringType"/>
    <xs:element name="monitoredAttributes" type="stringType"/>
    <xs:element name="proposedRepairActions" type="stringType"/>
    <xs:element name="additionalText" type="stringType"/>
    <xs:element name="additionalInformation" type="stringType"/>
  </xs:choice>
  <xs:attribute name="oper" type="operType" use="required"/>
</xs:complexType>
<!-- columnsType -->
<xs:complexType name="columnsType">
  <xs:sequence>
    <xs:element name="column" type="columnType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!-- filtersType -->
<xs:complexType name="filtersType">
  <xs:sequence>
    <xs:element name="filter" type="filterType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!-- rowType -->
<xs:complexType name="rowType">
  <xs:all>
    <xs:element name="id" type="stringType" minOccurs="0"/>
    <xs:element name="managedObjectClass" type="stringType" minOccurs="0"/>
    <xs:element name="managedObjectInstance" type="stringType" minOccurs="0"/>
    <xs:element name="location" type="stringType" minOccurs="0"/>
    <xs:element name="domain" type="stringType" minOccurs="0"/>
    <xs:element name="user_id" type="stringType" minOccurs="0"/>
    <xs:element name="eventTime" type="eventTimeType" minOccurs="0"/>
    <!-- datetime or undefined -->
    <xs:element name="cleared" type="eventTimeType" minOccurs="0"/>
    <xs:element name="probableCause" minOccurs="0">
      <xs:complexType>
        <xs:attribute name="value" type="alarmCauseType" use="required"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="eventType" minOccurs="0">
      <xs:complexType>
        <xs:attribute name="value" type="alarmEventTypeType" use="required"/>
      </xs:complexType>
    </xs:element>
    <!-- term -->
    <xs:element name="specificProblems" type="stringType" minOccurs="0"/>
    <xs:element name="perceivedSeverity" minOccurs="0">
      <xs:complexType>
        <xs:attribute name="value" type="severityType" use="required"/>
      </xs:complexType>
    </xs:element>
  </xs:all>
</xs:complexType>

```



```

<xs:element name="originalSeverity" minOccurs="0">
  <xs:complexType>
    <xs:attribute name="value" type="severityType" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="backedupStatus" minOccurs="0">
  <xs:complexType>
    <xs:attribute name="value" type="backedupStatusType" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="backupObject" type="stringType" minOccurs="0"/>
<xs:element name="trendIndication" minOccurs="0">
  <xs:complexType>
    <xs:attribute name="value" type="trendIndicationType" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="thresholdInformation" type="stringType" minOccurs="0"/>
<xs:element name="notificationIdentifier" type="stringType" minOccurs="0"/>
<xs:element name="correlatedNotifications" type="stringType" minOccurs="0"/>
<xs:element name="stateChangeDefinition" type="stringType" minOccurs="0"/>
<xs:element name="monitoredAttributes" type="stringType" minOccurs="0"/>
<xs:element name="proposedRepairActions" type="stringType" minOccurs="0"/>
<xs:element name="additionalText" type="pstrType" minOccurs="0"/>
<xs:element name="additionalInformation" type="stringType" minOccurs="0"/>
</xs:all>
</xs:complexType>
<!-- rowsType -->
<xs:complexType name="rowsType">
  <xs:sequence>
    <xs:element name="row" type="rowType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="columns" type="columnsType"/>
            <xs:element name="filters" type="filtersType"/>
          </xs:sequence>
          <xs:attribute name="mediator" type="xs:string" use="required"/>
          <xs:attribute name="type" type="requestType" use="required"/>
          <xs:attribute name="count" type="xs:integer" use="optional"/>
          <xs:attribute name="offset" type="xs:integer" use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```
</xs:element>
<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="time" type="eventTimeType"/>
            <xs:element name="columns" type="columnsType"/>
            <xs:element name="rows" type="rowsType"/>
          </xs:sequence>
          <xs:attribute name="type" type="requestType" use="required"/>
          <xs:attribute name="total" type="xs:integer" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>
```

rps.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- columnNameType -->
  <xs:simpleType name="columnNameType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="id"/>
      <xs:enumeration value="managedObjectClass"/>
      <xs:enumeration value="managedObjectInstance"/>
      <xs:enumeration value="location"/>
      <xs:enumeration value="domain"/>
      <xs:enumeration value="user_id"/>
      <xs:enumeration value="eventTime"/>
      <xs:enumeration value="cleared"/>
      <xs:enumeration value="probableCause"/>
      <xs:enumeration value="eventType"/>
      <xs:enumeration value="specificProblems"/>
      <xs:enumeration value="perceivedSeverity"/>
      <xs:enumeration value="originalSeverity"/>
      <xs:enumeration value="backedupStatus"/>
      <xs:enumeration value="backupObject"/>
      <xs:enumeration value="trendIndication"/>
      <xs:enumeration value="thresholdInformation"/>
      <xs:enumeration value="notificationIdentifier"/>
      <xs:enumeration value="correlatedNotifications"/>
      <xs:enumeration value="stateChangeDefinition"/>
      <xs:enumeration value="monitoredAttributes"/>
      <xs:enumeration value="proposedRepairActions"/>
      <xs:enumeration value="additionalText"/>
      <xs:enumeration value="additionalInformation"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- trendIndicationType -->
  <xs:simpleType name="trendIndicationType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="moreSevere"/>
      <xs:enumeration value="noChange"/>
      <xs:enumeration value="lessSevere"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- backedupStatusType -->
  <xs:simpleType name="backedupStatusType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="true"/>
      <xs:enumeration value="false"/>
      <xs:enumeration value="undefined"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- severityType -->

```

```

<xs:simpleType name="severityType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="cleared"/>
    <xs:enumeration value="indeterminate"/>
    <xs:enumeration value="critical"/>
    <xs:enumeration value="major"/>
    <xs:enumeration value="minor"/>
    <xs:enumeration value="warning"/>
  </xs:restriction>
</xs:simpleType>
<!-- alarmEventTypeType -->
<xs:simpleType name="alarmEventTypeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="other"/>
    <xs:enumeration value="communicationsAlarm"/>
    <xs:enumeration value="qualityOfServiceAlarm"/>
    <xs:enumeration value="processingErrorAlarm"/>
    <xs:enumeration value="equipmentAlarm"/>
    <xs:enumeration value="environmentalAlarm"/>
    <xs:enumeration value="integrityViolation"/>
    <xs:enumeration value="operationalViolation"/>
    <xs:enumeration value="physicalViolation"/>
    <xs:enumeration value="securityServiceOrMechanismViolation"/>
    <xs:enumeration value="timeDomainViolation"/>
  </xs:restriction>
</xs:simpleType>
<!-- alarmCauseType -->
<xs:simpleType name="alarmCauseType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="other"/>
    <xs:enumeration value="unexpectedInformation"/>
    <xs:enumeration value="unauthorizedAccessAttempt"/>
    <xs:enumeration value="proceduralError"/>
    <xs:enumeration value="outOfService"/>
    <xs:enumeration value="outOfHoursActivity"/>
    <xs:enumeration value="nonRepudiationFailure"/>
    <xs:enumeration value="keyExpired"/>
    <xs:enumeration value="informationOutOfSequence"/>
    <xs:enumeration value="informationModificationDetected"/>
    <xs:enumeration value="informationMissing"/>
    <xs:enumeration value="duplicateInformation"/>
    <xs:enumeration value="denialOfService"/>
    <xs:enumeration value="delayedInformation"/>
    <xs:enumeration value="cableTamper"/>
    <xs:enumeration value="breachOfConfidentiality"/>
    <xs:enumeration value="authenticationFailure"/>
    <xs:enumeration value="versionMismatchX733"/>
    <xs:enumeration value="underlyingResourceUnavailable"/>
    <xs:enumeration value="transmitterFailure"/>
    <xs:enumeration value="transmitFailureX733"/>
  </xs:restriction>
</xs:simpleType>

```

```
<xs:enumeration value="toxicLeakDetected"/>
<xs:enumeration value="timingProblemX733"/>
<xs:enumeration value="thresholdCrossed"/>
<xs:enumeration value="temperatureUnacceptable"/>
<xs:enumeration value="storageCapacityProblemX733"/>
<xs:enumeration value="softwareProgramError"/>
<xs:enumeration value="softwareProgramAbnormallyTerminated"/>
<xs:enumeration value="softwareErrorX733"/>
<xs:enumeration value="retransmissionRateExcessive"/>
<xs:enumeration value="responseTimeExcessive"/>
<xs:enumeration value="resourceAtOrNearingCapacity"/>
<xs:enumeration value="remoteNodeTransmissionErrorX733"/>
<xs:enumeration value="receiverFailureX733"/>
<xs:enumeration value="receiveFailureX733"/>
<xs:enumeration value="queueSizeExceeded"/>
<xs:enumeration value="pumpFailureX733"/>
<xs:enumeration value="processorProblems"/>
<xs:enumeration value="pressureUnacceptable"/>
<xs:enumeration value="powerProblems"/>
<xs:enumeration value="performanceDegraded"/>
<xs:enumeration value="outputDeviceError"/>
<xs:enumeration value="outOfMemoryX733"/>
<xs:enumeration value="multiplexerProblemX733"/>
<xs:enumeration value="materialSupplyExhausted"/>
<xs:enumeration value="lossOfSignalX733"/>
<xs:enumeration value="lossOfFrameX733"/>
<xs:enumeration value="localNodeTransmissionErrorX733"/>
<xs:enumeration value="leakDetected"/>
<xs:enumeration value="lanError"/>
<xs:enumeration value="inputDeviceError"/>
<xs:enumeration value="inputOutputDeviceError"/>
<xs:enumeration value="humidityUnacceptable"/>
<xs:enumeration value="heatingVentCoolingSystemProblem"/>
<xs:enumeration value="framingErrorX733"/>
<xs:enumeration value="fireDetected"/>
<xs:enumeration value="fileErrorX733"/>
<xs:enumeration value="excessiveVibration"/>
<xs:enumeration value="equipmentMalfunction"/>
<xs:enumeration value="enclosureDoorOpenX733"/>
<xs:enumeration value="dteDceInterfaceError"/>
<xs:enumeration value="degradedSignalX733"/>
<xs:enumeration value="dataSetOrModemError"/>
<xs:enumeration value="cpuCyclesLimitExceeded"/>
<xs:enumeration value="corruptData"/>
<xs:enumeration value="congestionX733"/>
<xs:enumeration value="configurationOrCustomizationError"/>
<xs:enumeration value="communicationsSubsystemFailure"/>
<xs:enumeration value="communicationsProtocolError"/>
<xs:enumeration value="callEstablishmentError"/>
<xs:enumeration value="bandwidthReducedX733"/>
```

```
<xs:enumeration value="applicationSubsystemFailure"/>
<xs:enumeration value="adapterError"/>
<xs:enumeration value="systemResourcesOverload"/>
<xs:enumeration value="reducedLoggingCapability"/>
<xs:enumeration value="excessiveRetransmissionRate"/>
<xs:enumeration value="excessiveResponseTime"/>
<xs:enumeration value="excessiveErrorRate"/>
<xs:enumeration value="congestion"/>
<xs:enumeration value="bandwidthReduced"/>
<xs:enumeration value="versionMismatch"/>
<xs:enumeration value="underlyingResourceUnavailable"/>
<xs:enumeration value="timeoutExpired"/>
<xs:enumeration value="softwareError"/>
<xs:enumeration value="outOfMemory"/>
<xs:enumeration value="fileError"/>
<xs:enumeration value="databaseInconsistency"/>
<xs:enumeration value="configurationOrCustomisationError"/>
<xs:enumeration value="applicationSubsystemFailure"/>
<xs:enumeration value="lossOfRealTime1"/>
<xs:enumeration value="sfwrDownloadFailure"/>
<xs:enumeration value="sfwrEnvironmentProblem"/>
<xs:enumeration value="outOfCPUCycles"/>
<xs:enumeration value="corruptData"/>
<xs:enumeration value="memoryMismatch"/>
<xs:enumeration value="storageCapacityProblem"/>
<xs:enumeration value="externalPointFailure"/>
<xs:enumeration value="externalEquipmentFailure"/>
<xs:enumeration value="coolingSystemFailure"/>
<xs:enumeration value="toxicGas"/>
<xs:enumeration value="smoke"/>
<xs:enumeration value="lowWater"/>
<xs:enumeration value="lowTemperatue"/>
<xs:enumeration value="lowCablePressure"/>
<xs:enumeration value="lowHumidity"/>
<xs:enumeration value="lowFuel"/>
<xs:enumeration value="intrusionDetection"/>
<xs:enumeration value="iceBuildUp"/>
<xs:enumeration value="highWind"/>
<xs:enumeration value="highTemperature"/>
<xs:enumeration value="highHumidity"/>
<xs:enumeration value="flood"/>
<xs:enumeration value="fire"/>
<xs:enumeration value="explosiveGas"/>
<xs:enumeration value="enclosureDoorOpen"/>
<xs:enumeration value="ventilationsSystemFailure"/>
<xs:enumeration value="rectifierLowFVoltage"/>
<xs:enumeration value="rectifierHighVoltage"/>
<xs:enumeration value="rectifierFailure"/>
<xs:enumeration value="pumpFailure"/>
<xs:enumeration value="lowBatteryThreshold"/>
```

```
<xs:enumeration value="generatorFailure"/>
<xs:enumeration value="fuseFailure"/>
<xs:enumeration value="fireDetectorFailure"/>
<xs:enumeration value="engineFailure"/>
<xs:enumeration value="coolingFanFailure"/>
<xs:enumeration value="commercialPowerFailure"/>
<xs:enumeration value="batteryFailure"/>
<xs:enumeration value="batteryDischarging"/>
<xs:enumeration value="airDryerFailure"/>
<xs:enumeration value="airConditioningFailure"/>
<xs:enumeration value="airCompressorFailure"/>
<xs:enumeration value="protectingResourceFailure"/>
<xs:enumeration value="protectionMechanismFailure"/>
<xs:enumeration value="tranceiverFailure"/>
<xs:enumeration value="signalQualityEvaluationFailure"/>
<xs:enumeration value="powerSupplyFailure"/>
<xs:enumeration value="lossOfRedundancy"/>
<xs:enumeration value="lossOfSynchronisation"/>
<xs:enumeration value="iODeviceError"/>
<xs:enumeration value="frequencyHoppingFailure"/>
<xs:enumeration value="diskFailure"/>
<xs:enumeration value="batteryChargingFailure"/>
<xs:enumeration value="antennaFailure"/>
<xs:enumeration value="realTimeClockFailure"/>
<xs:enumeration value="replaceableUnitProblem"/>
<xs:enumeration value="trunkCardProblem"/>
<xs:enumeration value="transmitterFailure"/>
<xs:enumeration value="timingProblem"/>
<xs:enumeration value="terminalProblem"/>
<xs:enumeration value="synchronizationSourceMismatch"/>
<xs:enumeration value="replaceableUnitTypeMismatch"/>
<xs:enumeration value="replaceableUnitMissing"/>
<xs:enumeration value="receiverFailure"/>
<xs:enumeration value="protectionPathFailure"/>
<xs:enumeration value="processorProblem"/>
<xs:enumeration value="powerProblem"/>
<xs:enumeration value="nEIdentifierDuplication"/>
<xs:enumeration value="multiplexerProblem"/>
<xs:enumeration value="lineCardProblem"/>
<xs:enumeration value="externalIFDeviceProblem"/>
<xs:enumeration value="equipmentIdentifierDuplication"/>
<xs:enumeration value="dataSetProblem"/>
<xs:enumeration value="backplaneFailure"/>
<xs:enumeration value="routingFailure"/>
<xs:enumeration value="remoteNodeTransmissionError"/>
<xs:enumeration value="localNodeTransmissionError"/>
<xs:enumeration value="invalidMessageReceived"/>
<xs:enumeration value="connectionEstablishmentError"/>
<xs:enumeration value="broadcastChannelFailure"/>
<xs:enumeration value="demodulationFailure"/>
```

```

    <xs:enumeration value="modulationFailure"/>
    <xs:enumeration value="transmitFailure"/>
    <xs:enumeration value="receiveFailure"/>
    <xs:enumeration value="lossOfMultiFrame"/>
    <xs:enumeration value="signalLabelMismatch"/>
    <xs:enumeration value="unavailable"/>
    <xs:enumeration value="pathTraceMismatch"/>
    <xs:enumeration value="excessiveBER"/>
    <xs:enumeration value="remoteAlarmInterface"/>
    <xs:enumeration value="transmissionError"/>
    <xs:enumeration value="payloadTypeMismatch"/>
    <xs:enumeration value="lossOfSignal"/>
    <xs:enumeration value="lossOfPointer"/>
    <xs:enumeration value="lossOfFrame"/>
    <xs:enumeration value="framingError"/>
    <xs:enumeration value="farEndReceiverFailure"/>
    <xs:enumeration value="degradedSignal"/>
    <xs:enumeration value="callSetUpFailure"/>
    <xs:enumeration value="aIS"/>
  </xs:restriction>
</xs:simpleType>
<!-- eventType -->
<xs:complexType name="eventType">
  <xs:attribute name="year" type="xs:integer" use="required"/>
  <xs:attribute name="month" type="xs:integer" use="required"/>
  <xs:attribute name="day" type="xs:integer" use="required"/>
  <xs:attribute name="hour" type="xs:integer" use="required"/>
  <xs:attribute name="minute" type="xs:integer" use="required"/>
  <xs:attribute name="second" type="xs:integer" use="required"/>
</xs:complexType>
<!-- alarmEventType -->
<xs:complexType name="alarmEventType">
  <xs:attribute name="cause" type="alarmCauseType" use="required"/>
  <xs:attribute name="type" type="alarmEventTypeType" use="required"/>
  <xs:attribute name="severity" type="severityType" use="required"/>
  <xs:attribute name="trend" type="trendIndicationType" use="required"/>
  <xs:attribute name="text" type="xs:string" use="required"/>
</xs:complexType>
<!-- rpsEventType -->
<xs:complexType name="rpsEventType">
  <xs:sequence>
    <xs:element name="time" type="eventType"/>
    <xs:element name="info" type="alarmEventType"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required"/>
  <xs:attribute name="class" type="xs:string" use="required"/>
  <xs:attribute name="location" type="xs:string" use="required"/>
  <xs:attribute name="instance" type="xs:string" use="required"/>
</xs:complexType>
<!-- events -->

```



```

<xs:element name="events">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="event" type="rpsEventType" minOccurs="0" maxOccurs="unbounded"/
    >
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/alarms_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_alarms_list.xsd">
  <request offset="0" mediator="md1" type="all" count="5">
    <columns>
      <column name="id"/>
      <column name="originalSeverity"/>
      <column name="perceivedSeverity"/>
      <column name="managedObjectClass"/>
      <column name="managedObjectInstance"/>
      <column name="location"/>
      <column name="user_id"/>
      <column sort="desc" name="eventTime"/>
      <column name="probableCause"/>
      <column name="eventType"/>
      <column name="additionalText"/>
    </columns>
    <filters>
      <filter oper="more_or_equal">
        <eventTime minute="0" second="0" day="17" month="09" year="2017" hour="0" />
      </filter>
      <filter oper="not_equal">
        <id value="65353061-3434-3734-6435-383562323062"/>
      </filter>
    </filters>
  </request>
</in>

```

Ответ:

200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alarms_list.xsd">
  <response type="all" total="291">
    <time year="2019" month="5" day="14" hour="1" minute="48" second="5"/>
    <columns>
      <column name="id"/>
      <column name="originalSeverity"/>
      <column name="perceivedSeverity"/>
      <column name="managedObjectClass"/>
      <column name="managedObjectInstance"/>
      <column name="location"/>
      <column name="user_id"/>
      <column name="eventTime" sort="desc"/>
      <column name="probableCause"/>
      <column name="eventType"/>
      <column name="additionalText"/>
    </columns>
    <rows>
      <row>
        <id value="30363135-6331-6332-3262-326333346530"/>
        <originalSeverity value="critical"/>
        <perceivedSeverity value="cleared"/>
        <managedObjectClass value="ecss::cluster"/>
        <managedObjectInstance value="sip1"/>
        <location value="mycelium1@ecss1"/>
        <user_id value="[system]"/>
        <eventTime year="2019" month="5" day="14" hour="0" minute="26" second="48"/>
        <probableCause value="softwareProgramAbnormallyTerminated"/>
        <eventType value="processingErrorAlarm"/>
        <additionalText id="rps_tring_cluster" template="Cluster ~s lost" value="Cluster
"sip1" lost">
          <arg value=""sip1""/>
        </additionalText>
      </row>
      <row>
        <id value="30363135-6331-6332-3433-646462386439"/>
        <originalSeverity value="major"/>
        <perceivedSeverity value="cleared"/>
        <managedObjectClass value="ecss::cluster::node"/>
        <managedObjectInstance value="sip1@ecss1"/>
        <location value="ds1@ecss1"/>
        <user_id value="[system]"/>
        <eventTime year="2019" month="5" day="14" hour="0" minute="26" second="48"/>
        <probableCause value="softwareProgramAbnormallyTerminated"/>
        <eventType value="processingErrorAlarm"/>
        <additionalText id="rps_tring_node_cluster" template="Node ~s of ~s cluster
terminated abnormally" value="Node sip1@ecss1 of "sip1" cluster terminated abnormally">

```

```

        <arg value="sip1@ecss1"/>
        <arg value=""sip1""/>
    </additionalText>
</row>
<row>
    <id value="30363135-6331-6332-3261-653539663736"/>
    <originalSeverity value="critical"/>
    <perceivedSeverity value="cleared"/>
    <managedObjectClass value="ecss::cluster"/>
    <managedObjectInstance value="md1"/>
    <location value="mycelium1@ecss1"/>
    <user_id value="[system]"/>
    <eventTime year="2019" month="5" day="14" hour="0" minute="26" second="14"/>
    <probableCause value="softwareProgramAbnormallyTerminated"/>
    <eventType value="processingErrorAlarm"/>
    <additionalText id="rps_tring_cluster" template="Cluster ~s lost" value="Cluster
"md1" lost">
        <arg value=""md1""/>
    </additionalText>
</row>
<row>
    <id value="30363135-6331-6332-3334-396461313836"/>
    <originalSeverity value="major"/>
    <perceivedSeverity value="cleared"/>
    <managedObjectClass value="ecss::cluster::node"/>
    <managedObjectInstance value="md1@ecss1"/>
    <location value="sip1@ecss1"/>
    <user_id value="[system]"/>
    <eventTime year="2019" month="5" day="14" hour="0" minute="26" second="14"/>
    <probableCause value="softwareProgramAbnormallyTerminated"/>
    <eventType value="processingErrorAlarm"/>
    <additionalText id="rps_tring_node_cluster" template="Node ~s of ~s cluster
terminated abnormally" value="Node md1@ecss1 of "md1" cluster terminated abnormally">
        <arg value="md1@ecss1"/>
        <arg value=""md1""/>
    </additionalText>
</row>
<row>
    <id value="30363135-6331-6332-3239-313933633437"/>
    <originalSeverity value="critical"/>
    <perceivedSeverity value="cleared"/>
    <managedObjectClass value="ecss::cluster"/>
    <managedObjectInstance value="core1"/>
    <location value="mycelium1@ecss1"/>
    <user_id value="[system]"/>
    <eventTime year="2019" month="5" day="14" hour="0" minute="26" second="14"/>
    <probableCause value="softwareProgramAbnormallyTerminated"/>
    <eventType value="processingErrorAlarm"/>
    <additionalText id="rps_tring_cluster" template="Cluster ~s lost" value="Cluster
"core1" lost">

```

```
        <arg value=""core1""/>
      </additionalText>
    </row>
  </rows>
</response>
</out>
```

hc_alarms_mask_list

Команда возвращает список масок, используемых для маскирования алармов.

Аналог команды в CoCon:

```
cluster/mediator/<md>/alarms/masklist
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alarms_mask_list

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

hc_alarms_mask_common.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- stringType -->
  <xs:complexType name="stringType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- stringArrayType -->
  <xs:complexType name="stringArrayType">
    <xs:sequence>
      <xs:element name="item" type="stringType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- alarmMaskType -->
  <xs:complexType name="alarmMaskType">
    <xs:sequence>
      <xs:choice>
        <xs:element name="location" type="stringType"/>
        <xs:element name="ignoreLocations" type="stringArrayType"/>
      </xs:choice>
      <xs:choice>
        <xs:element name="class" type="stringType"/>
        <xs:element name="ignoreClasses" type="stringArrayType"/>
      </xs:choice>
      <xs:choice>
        <xs:element name="instance" type="stringType"/>
        <xs:element name="ignoreInstances" type="stringArrayType"/>
      </xs:choice>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="optional"/>
    <xs:attribute name="enabled" type="xs:boolean" use="required"/>
    <xs:attribute name="date" type="xs:string" use="optional"/>
  </xs:complexType>
</xs:schema>

```

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="hc_alarms_mask_common.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="mediator" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="masks">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="mask" type="alarmMaskType" minOccurs="0"
maxOccurs="unbounded"
"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/alarms_mask_list

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_alarms_mask_list.xsd">
  <request mediator="md1"/>
</in>

```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alarms_mask_list.xsd">
  <mask>
    <mask id="30363135-6334-6262-3230-633937383736" enabled="true" date="14.05.2019
10:50:10">
      <location value="*" />
      <class value="*" />
      <instance value="ds1@ecss1" />
    </mask>
  </mask>
</out>
```

hc_node_rps_alarms

Команда позволяет включить/выключить систему мониторинга и вывода аварий.

Аналог команды в CoCon:

```
/cluster/mediator/<CLUSTER>/properties/rpss/set
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/node_rps_alarms

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name = "stateType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="on"/>
      <xs:enumeration value="off"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="state" type="stateType" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="node" type="nodeType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/node_rps_alarms

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="node_rps_onoff.xsd">
  <node name="core1@ecss1" state="on" />
</in>

```

Ответ:

201

[Hc_alarms_mask_add](#)

Команда позволяет создать маску/маски для аварий.

Аналог команды в CoCon:

```
/cluster/mediator/<CLUSTER>/alarms/maskadd
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alarms_mask_add

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

[hc_alarms_mask_common.xsd](#)

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="hc_alarms_mask_common.xsd"/>
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:sequence>
      <xs:element name="mask" type="alarmMaskType" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
  <!-- errorReasonType -->
  <xs:simpleType name="errorReasonType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="conflict" />
      <xs:enumeration value="notfound" />
    </xs:restriction>
  </xs:simpleType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:sequence>
      <xs:element name="mask" type="alarmMaskType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="conflictMask" type="alarmMaskType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="reason" type="errorReasonType" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="masks">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="mask" type="alarmMaskType"
minOccurs="0" maxOccurs="unbounded"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="mediator" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->

```

```

<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="ok" type="okType" minOccurs="0"
maxOccurs="unbounded"/>
            <xs:element name="error" type="errorType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/alarms_mask_add

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_alarms_mask_add.xsd">
  <request mediator="md1">
    <mask>
      <mask enabled="true">
        <location value="core1@ecss1"/>
        <class value="ecss::zmq::connection"/>
        <instance value="msr_2(192.168.2.22:5700)"/>
      </mask>
      <mask enabled="true">
        <location value="ecss1"/>
        <class value="hw::interfaces"/>
        <instance value="bond1.2:2"/>
      </mask>
    </mask>
  </request>
</in>

```

Ответ:

Код: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alarms_mask_add.xsd">
  <response>
    <ok>
      <mask id="30363135-6336-3132-3861-303135366466" enabled="true" date="14.05.2019
12:21:44">
        <location value="ecss1"/>
        <class value="hw::interfaces"/>
        <instance value="bond1.2:2"/>
      </mask>
    </ok>
    <ok>
      <mask id="30363135-6336-3132-3861-303038366132" enabled="true" date="14.05.2019
12:21:44">
        <location value="core1@ecss1"/>
        <class value="ecss::zmq::connection"/>
        <instance value="msr_2(192.168.2.22:5700)"/>
      </mask>
    </ok>
  </response>
</out>

```

hc_alarms_mask_del

Команда позволяет удалить маску/маски для аварий.

Аналог команды в CoCon:

```
/cluster/mediator/<CLUSTER>/alarms/maskdel
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alarms_mask_del

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить двумя сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- alarmMaskType -->
  <xs:complexType name="alarmMaskType">
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="masks">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="mask" type="alarmMaskType"
minOccurs="0" maxOccurs="unbounded"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="mediator" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/alarms_mask_del

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_alarms_mask_del.xsd">
  <request mediator="md1">
    <masks>
      <mask id="65353163-6466-6334-3861-666566626238"/>
      <mask id="65353163-3738-3161-3636-646462383537"/>
    </masks>
  </request>
</in>

```

Ответ:

204

[Hc_alarms_clear](#)

Команда позволяет удалить аварии из системы по идентификатору.

Аналог команды в CoCon:

```
/cluster/mediator/<CLUSTER>/alarms/clear
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alarms_clear

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- clearEventType -->
  <xs:complexType name="clearEventType">
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="events">
          <xs:complexType>
            <xs:choice>
              <xs:sequence>
                <xs:element name="event" type="clearEventType"
maxOccurs="unbounded"/>
              </xs:sequence>
              <xs:element name="all"/>
            </xs:choice>
            <xs:attribute name="mediator" type="xs:string" use="required"/>
            <xs:attribute name="userId" type="xs:string" use="optional">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  Deprecated attribute.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

rps.xsd:

См. схему запроса списка алармов в системе

Пример 1:

Отметка каждого из выбранных алармов, как просмотренные.

Запрос: http://192.168.1.21:9999/cluster/mediator/md/alarms_clear

Параметры:

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_alarms_clear.xsd">
  <events mediator="md1" userId="admin">
    <event id="30363333-3436-6564-3932-353930383363"/>
  </events>
</in>
```

Ответ:

204

Пример 2:

Отметка всех алармов, как просмотренные.

Запрос: http://192.168.1.21:9999/cluster/mediator/md/alarms_clear

Параметры:

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_alarms_clear.xsd">
  <events mediator="md1" userId="ecss1">
    <all/>
  </events>
</in>
```

Ответ:

204

hc_alarms_delete

Команда позволяет удалить очищенные аварии из системы по следующим критериям:

- последние N алармов;
- в случае, если после удаления остаются активные алармы, отображает их количество;
- алармы за определенный промежуток времени;
- алармы с определенными идентификаторами;
- все алармы.

Аналог команды в CoCon:

```
/cluster/mediator/<CLUSTER>/alarms/delete
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alarms_delete

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- timeType -->
  <xs:complexType name="timeType">
    <xs:attribute name="year" type="xs:integer" use="required"/>
    <xs:attribute name="month" type="xs:integer" use="required"/>
    <xs:attribute name="day" type="xs:integer" use="required"/>
    <xs:attribute name="hour" type="xs:integer" use="required"/>
    <xs:attribute name="minute" type="xs:integer" use="required"/>
    <xs:attribute name="second" type="xs:integer" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:choice>
              <xs:element name="all">
                <xs:complexType/>
              </xs:element>
              <xs:element name="last">
                <xs:complexType>
                  <xs:attribute name="count" type="xs:integer" use="required"/>
                </xs:complexType>
              </xs:element>
              <xs:element name="between">
                <xs:complexType>
                  <xs:all>
                    <xs:element name="from" type="timeType" minOccurs="0"
maxOccurs="1"/>
                    <xs:element name="to" type="timeType" minOccurs="0"
maxOccurs="1"/>
                  </xs:all>
                </xs:complexType>
              </xs:element>
            </xs:choice>
          </xs:complexType>
        </xs:element>
        <xs:element name="ids">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="id" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="value" type="xs:string"
use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

        </xs:choice>
        <xs:attribute name="mediator" type="xs:string" use="required"/>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="result">
                <xs:complexType>
                    <xs:attribute name="active_alarms_count" type="xs:integer" use="required"
/>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

1. Пример запроса удаления последних 10-ти алармов:
Запрос: http://192.168.1.21:9999/commands/alarms_delete
Параметры:

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_alarms_delete.xsd">
    <request mediator="md1">
        <last count="10" />
    </request>
</in>

```

Ответ: 204

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_alarms_delete.xsd">
    <result active_alarms_count="1"/>
</out>

```

2. Пример запроса удаления алармов, случившихся после 2019/05/13 16:00:00:
Запрос: http://192.168.1.21:9999/commands/alarms_delete
Параметры:

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_alarms_delete.xsd">
  <request mediator="md1">
    <between>
      <from day="13" month="5" year="2019" hour="16" minute="0" second="0" />
    </between>
  </request>
</in>
```

Ответ: 204

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alarms_delete.xsd">
  <result active_alarms_count="3"/>
</out>
```

3. Пример запроса удаления алармов по идентификаторам:

Запрос: http://192.168.1.21:9999/commands/alarms_delete

Параметры:

Ответ: 200

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_alarms_delete.xsd">
  <request mediator="md1">
    <ids>
      <id value="30363333-3932-6164-3534-303931383665" />
    </ids>
  </request>
</in>
```

4. Пример запроса удаления всех алармов:

Запрос: http://192.168.1.21:9999/commands/alarms_delete

Параметры:

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_alarms_delete.xsd">
  <request mediator="md1">
    <all/>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?><out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_alarms_delete.xsd"><result active_alarms_count="0"/></
out>
```

hc_alarm_notifiers_options_clean

Команда позволяет сбрасывать значения параметров службы уведомления по электронной почте или Jabber.

Аналог команды в CoCon:

```
/cluster/mediator/<CLUSTER>/alarms/notifiers/email/clean
/domain/<DOMAIN>/alarms/notifiers/email/clean
```

```
/cluster/mediator/<CLUSTER>/alarms/notifiers/jabber/clean
/domain/<DOMAIN>/alarms/notifiers/jabber/clean
```

Шаблон HTTP-запроса:

http://host:port/commands/alarm_notifiers_options_clean

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XML-схема

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean properties of alarm notifier service.
      Usage: http://server:9999/commands/alarm_notifiers_options_clean.
      Implementation: hc_alarm_notifiers_options_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

http://192.168.1.21:9999/commands/alarm_notifiers_options_clean

Запрос:

```
<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alarm_notifiers_options_clean.xsd">
  <request>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <option name="email_body_limit"/>
        <option name="email_enabled"/>
        <option name="jabber_body_format"/>
        <option name="jabber_body_limit"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alarm_notifiers_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="email_body_limit"/>
        <success option="email_enabled"/>
        <success option="jabber_body_format"/>
        <success option="jabber_body_limit"/>
      </domain>
    </domain_options>
  </result>
</out>
```

[Hc_alarm_notifiers_options_info](#)

Команда просмотра текущих настроек аварийных уведомлений, которые передаются через электронную почту или Jabber.

Аналог команды в CoCon:

```
/cluster/mediator/<CLUSTER>/alarms/notifiers/email/info  
/domain/<DOMAIN>/alarms/notifiers/email/info
```

```
/cluster/mediator/<CLUSTER>/alarms/notifiers/jabber/info  
/domain/<DOMAIN>/alarms/notifiers/jabber/info
```

Шаблон HTTP-запроса:

http://host:port/commands/alarm_notifiers_options_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XML-схема


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of alarm notifier service.
      Usage: http://server:9999/commands/alarm\_notifiers\_options\_info.
      Implementation: hc_alarm_notifiers_options_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <xs:complexType name="alarmNotifiersOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="email_body_format" type="xs:string" use="optional"/>
        <xs:attribute name="email_body_limit" type="xs:string" use="optional"/>
        <xs:attribute name="email_enabled" type="booleanType" use="optional"/>
        <xs:attribute name="email_from_name" type="xs:string" use="optional"/>
        <xs:attribute name="email_subject_format" type="xs:string" use="optional"/>
        <xs:attribute name="email_subject_limit" type="xs:string" use="optional"/>
        <xs:attribute name="to_emails" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              The list of email addresses separated by ";".
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="email_to_name" type="xs:string" use="optional"/>
        <xs:attribute name="jabber_body_format" type="xs:string" use="optional"/>
        <xs:attribute name="jabber_body_limit" type="xs:string" use="optional"/>
        <xs:attribute name="jabber_enabled" type="booleanType" use="optional"/>
        <xs:attribute name="to_jids" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              The list of jabber addresses separated by ";".
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>

```

```

        </xs:all>
    </xs:complexType>
    <!-- in -->
    <xs:element name="in">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="request">
                    <xs:complexType>
                        <xs:all>
                            <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1"/>
                        </xs:all>
                    </xs:complexType>
                </xs:element>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <!-- out -->
    <xs:element name="out">
        <xs:complexType>
            <xs:choice>
                <xs:element name="result" type="resultType"/>
                <xs:element name="error" type="errorType"/>
            </xs:choice>
        </xs:complexType>
    </xs:element>
</xs:schema>

```

Пример:

http://192.168.1.21:9999/commands/alarm_notifiers_options_info

Запрос:

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_alarm_notifiers_options_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>

```

Ответ:

200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alarm_notifiers_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="alarmNotifiersOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="alarmNotifiersOptionsType" to_jids=""
to_emails="ryaskowa.s@yandex.ru" jabber_enabled="false" jabber_body_limit="1000"
jabber_body_format="Class: %CLASS% ~nLocation: %LOCATION% ~nInstance: %INSTANCE% ~nTime: %TIME%
~nText: %TEXT%" email_to_name="ECSS-SUPPORT" email_subject_limit="75" email_subject_format="ALARM
(%SEVERITY%)" email_from_name="ECSS-NO-REPLY-ALARM-NOTIFICATOR" email_enabled="true"
email_body_limit="1500" email_body_format="Class: %CLASS% ~nLocation: %LOCATION% ~nInstance:
%INSTANCE% ~nTime: %TIME% ~nText: %TEXT%"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Hc_alarm_notifiers_options_set

Команда настройки параметров уведомления, отправляемого по электронной почте или Jabber.

Аналог команды в CoCon:

```

/ccluster/mediator/<CLUSTER>/alarms/notifiers/email/set
/domain/<DOMAIN>/alarms/notifiers/email/set

```

```

/ccluster/mediator/<CLUSTER>/alarms/notifiers/jabber/set
/domain/<DOMAIN>/alarms/notifiers/jabber/set

```

Шаблон HTTP-запроса:

http://host:port/commands/alarm_notifiers_options_set

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XML-схема

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set properties of alarm notifier service.
      Usage: http://server:9999/commands/alarm\_notifiers\_options\_set.
      Implementation: hc_alarm_notifiers_options_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- Types -->
  <xs:complexType name="alarmNotifiersOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="email_body_format" type="xs:string" use="optional"/>
        <xs:attribute name="email_body_limit" type="xs:string" use="optional"/>
        <xs:attribute name="email_enabled" type="booleanType" use="optional"/>
        <xs:attribute name="email_from_name" type="xs:string" use="optional"/>
        <xs:attribute name="email_subject_format" type="xs:string" use="optional"/>
        <xs:attribute name="email_subject_limit" type="xs:string" use="optional"/>
        <xs:attribute name="to_emails" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              The list of email addresses separated by ";".
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="email_to_name" type="xs:string" use="optional"/>
        <xs:attribute name="jabber_body_format" type="xs:string" use="optional"/>
        <xs:attribute name="jabber_body_limit" type="xs:string" use="optional"/>
        <xs:attribute name="jabber_enabled" type="booleanType" use="optional"/>
        <xs:attribute name="to_jids" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              The list of jabber addresses separated by ";".
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>

```

```

        <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
</xs:complexType>
<!-- in -->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainSetInType"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
            <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

http://192.168.1.21:9999/commands/alarm_notifiers_options_set

Запрос:

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alarm_notifiers_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="alarmNotifiersOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="alarmNotifiersOptionsType" email_body_limit="1500"
email_enabled="false" email_from_name="ECSS-NO-REPLY-ALARM-NOTIFICATOR" email_subject_limit="75"
to_emails="ryaskowa.s@yandex.ru"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:

200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alarm_notifiers_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="email_body_limit"/>
        <success option="email_enabled"/>
        <success option="email_from_name"/>
        <success option="email_subject_limit"/>
        <success option="to_emails"/>
      </domain>
    </domain_options>
  </result>
</out>

```

[Hc_alarms_notifiers_send_test_msg](#)

Команда позволяет послать тестовое сообщение об аварии на ECSS-10 по почте/jabber.

Аналог команды в CoCon:

```

cluster/mediator/<CLUSTER>/alarms/notifiers/email/send_test_email
domain/<DOMAIN>/alarms/notifiers/email/send_test_email

```

```
cluster/mediator/<CLUSTER>/alarms/notifiers/jabber/send_test_email  
domain/<DOMAIN>/alarms/notifiers/jabber/send_test_email
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alarms_notifiers_send_test_msg

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />
  <xs:simpleType name="typeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="email"/>
      <xs:enumeration value="jabber"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="type" type="typeType" use="required"/>
            <xs:attribute name="msg" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!--Positive case-->
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <!--Or error-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/alarms_notifiers_send_test_msg


```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_alarms_notifiers_send_test.xsd">
  <request domain="biysk.local" type="email" msg="Test message"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alarms_notifiers_send_test.xsd">
  <response>
    <ok/>
  </response>
</out>
```

hc_alias_profile - Управление профилями алиасов

- [hc_alias_profile_list](#)
- [hc_alias_profile_declare](#)
- [hc_alias_profile_remove](#)
- [hc_alias_profile_info](#)
- [hc_alias_profile_set](#)
- [hc_alias_profile_clean](#)

hc_alias_profile_list

Команда возвращает список профилей алиасов, определенных в системе.

Аналог команды в CoCon:

domain/<DOMAIN>/alias/profiles/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_profile_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом:

Описание, общих для профилей алиасов, типов (**alias_profile_common.xsd**):

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <!-- attribute type used only to backward compatability with alias info.
      It's always equals with "profile"-->
    <xs:attribute name="type" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- propertyNameType-->
  <xs:complexType name="propertyNameType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- aliasProfileType-->
  <xs:complexType name="aliasProfileType">
    <xs:sequence>
      <xs:element name="properties">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>
</xs:schema>

```

alias_profile_list.xsd :

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="alias_profile_common.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="profiles">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="profile" type="aliasProfileType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/alias_profile_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="alias_profile_list.xsd">
  <request domain="biysk.local"/>
</in>

```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="alias_profile_list.xsd">
  <profiles>
    <profile domain="biysk.local" name="user_default" description="Default user profile">
      <properties/>
    </profile>
  </profiles>
</out>
```

[Hc_alias_profile_declare](#)

Команда создает новый профиль алиаса. Если не получается создать профиль, возвращает ошибку.

Аналог команды в CoCon:

domain/<DOMAIN>/alias/profiles/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_profile_declare

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом:

alias_profile_common.xsd: Hc_alias_profile_list.

alias_profile_declare.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="alias_profile_common.xsd"/>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="profile" type="aliasProfileType" minOccurs="1"
maxOccurs="1"/>
            </xs:sequence>
            <xs:attribute name="storage" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

http://192.168.1.21:9999/commands/alias_profile_declare

Запрос:

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="alias_profile_declare.xsd">
  <request storage="ds1">
    <profile name="front-office" description="for managers" domain="biysk.local">
      <properties/>
    </profile>
  </request>
</in>

```

Ответ:

201

Ответ 2 (ошибка):404

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="common_error.xsd">
  <error cmd="alias_profile_declare" reason="error" body="{error, 'Declare profile error:
{already_exists, {\"biysk.local\", \"front-office\"}}}" />
</out>
```

[Hc_alias_profile_remove](#)

Команда удаляет профили алиасов из определенного домена.

Аналог команды в CoCon:

domain/<DOMAIN>/alias/profiles/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_profile_remove

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом:

alias_profile_common.xsd: Hc_alias_profile_list.

alias_profile_remove.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="alias_profile_common.xsd"/>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="profiles">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="profile" type="aliasProfileType"
minOccurs="1" maxOccurs="unbounded"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:all>
            <xs:attribute name="storage" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/alias_profile_remove

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="alias_profile_remove.xsd">
  <request storage="ds1">
    <profiles>
      <profile name="front-office" domain="biysk.local">
        <properties/>
      </profile>
    </profiles>
  </request>
</in>

```

Ответ:

204

Нс_alias_profile_info

Команда получает список свойств профиля алиаса.

Аналог команды в CoCon:

`domain/<DOMAIN>/alias/profiles/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/alias_profile_info`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом:

`alias_profile_common.xsd`: Нс_alias_profile_list.

`alias_profile_info.xsd`:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="alias_profile_common.xsd"/>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="optional"/>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="profile_name" type="xs:string" use="required"/>
            <xs:attribute name="property_name" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="profiles">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="profile" type="aliasProfileType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

Запрос:

http://192.168.1.21:9999/commands/alias_profile_info

```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="alias_profile_info.xsd">
  <request domain="biysk.local" profile_name="front-office"/>
</in>

```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="alias_profile_info.xsd">
  <profiles>
    <profile domain="biysk.local" name="front-office" description="for managers">
      <properties/>
    </profile>
  </profiles>
</out>
```

Пример 2:

Запрос:

http://192.168.1.21:9999/commands/alias_profile_info

```
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="alias_profile_info.xsd">
  <request domain="biysk.local" profile_name="user_default" property_name="category"/>
</in>
```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="alias_profile_info.xsd">
  <profiles>
    <profile domain="biysk.local" name="user_default" description="">
      <properties>
        <property name="category" type="profile" value="10"/>
      </properties>
    </profile>
  </profiles>
</out>
```

[Hc_alias_profile_set](#)

Команда изменяет имеющиеся/добавляет новые свойства профиля алиаса.

Аналог команды в CoCon:

domain/**<DOMAIN>**/**alias/profiles/set**

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_profile_set

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом:

alias_profile_common.xsd: Hc_alias_profile_list.

alias_profile_set.xsd:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="alias_profile_common.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="properties" minOccurs="1" maxOccurs="1">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="property" type="propertyType"
minOccurs="0" maxOccurs="unbounded"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
          <xs:attribute name="storage" type="xs:string" use="optional"/>
          <xs:attribute name="domain" type="xs:string" use="required"/>
          <xs:attribute name="profile_name" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>
```

Пример:

Запрос:

http://192.168.1.21:9999/commands/alias_profile_set

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="alias_profile_set.xsd">
  <request domain="biysk.local" profile_name="front-office">
    <properties>
      <property name="category" type="profile" value="testCall"/>
      <property name="provider" type="profile" value="6"/>
    </properties>
  </request>
</in>
```

Ответ:

201

[Hc_alias_profile_clean](#)

Команда удаляет определенные свойства (если они указаны) профиля или все свойства профиля.

Аналог команды в CoCon:

domain/<DOMAIN>/alias/profiles/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_profile_clean

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом:

alias_profile_common.xsd: Hc_alias_profile_list.

alias_profile_clean.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="alias_profile_common.xsd"/>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="properties" minOccurs="1" maxOccurs="1">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="property" type="propertyNameType"
minOccurs="0" maxOccurs="unbounded"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="storage" type="xs:string" use="optional"/>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="profile_name" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

Запрос на удаление свойств provider, category из профиля front-office в домене biysk.local.

Запрос: http://192.168.1.21:9999/commands/alias_profile_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="alias_profile_clean.xsd">
  <request domain="biysk.local" profile_name="front-office">
    <properties>
      <property name="provider"/>
      <property name="category"/>
    </properties>
  </request>
</in>

```

Ответ:

204

Пример 2:

Запрос на удаление всех свойств из профиля front-office в домене biysk.local.

Запрос: http://192.168.1.21:9999/commands/alias_profile_clean

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="alias_profile_clean.xsd">
  <request domain="biysk.local" profile_name="front-office">
    <properties/>
  </request>
</in>
```

Ответ:

204

Ис_aliases - Управление списками алиасов

- [Общая XSD-схема](#)
- [Ис_aliases_list](#)
- [Ис_aliases_list2](#)

Общая XSD-схема

XSD-схема XML-файла запроса/ответа aliases.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- propertyTypeType-->
  <xs:simpleType name="propertyTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="global"/>
      <xs:enumeration value="domain"/>
      <xs:enumeration value="domain_interface"/>
      <xs:enumeration value="domain_address"/>
      <xs:enumeration value="address"/>
      <xs:enumeration value="profile"/>
      <xs:enumeration value="alias"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="propertyTypeType" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          For other type of properties value will be formatted specifically.
          Key:   timetables\TIMETABLE_NAME
          Value: {"name":"TIMETABLE_NAME", "filters":[{"date":"DAYS_LIST_OR_DAY_TYPE_N
AME","from":"FROM_HH:FROM_MM","to":"TO_HH:TO_MM"}],"properties":{"description | KEY":"VALUE"}}
          Sample: {"name":"tt2","filters":[{"date":"1 2 3 4 5","from":"9:0","to":"12:59
"}, {"date":"HOLIDAY","from":"13:0","to":"17:59"}],"properties":{"description":"Мой 2","key3":{"he
llo,world}}}}
          In case of strong security restrictions password properties may be shown as
          *****
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- aliasType-->
  <xs:complexType name="aliasType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="iface" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- addressType-->
  <xs:complexType name="addressType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

```

```

<!-- ifaceType-->
<xs:complexType name="ifaceType">
  <xs:sequence>
    <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>
<!-- errorType-->
<xs:complexType name="errorType">
  <xs:attribute name="iface" type="xs:string" use="required"/>
  <xs:attribute name="address" type="xs:string" use="required"/>
  <xs:attribute name="value" type="xs:string" use="required"/>
</xs:complexType>
<!-- Properties elements-->
<xs:complexType name="ifacesType">
  <xs:sequence>
    <xs:element name="iface" type="ifaceType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="domain" type="xs:string" use="required"/>
</xs:complexType>
<xs:complexType name="addressesType">
  <xs:sequence>
    <xs:element name="address" type="addressType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="domain" type="xs:string" use="required"/>
</xs:complexType>
<xs:complexType name="aliasesType">
  <xs:sequence>
    <xs:element name="alias" type="aliasType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="domain" type="xs:string" use="required"/>
</xs:complexType>
<xs:complexType name="domainType">
  <xs:sequence>
    <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="aliases" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="domain" type="xs:string" use="required"/>
          <xs:attribute name="addr" type="xs:string" use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```



```

        <xs:attribute name="iface" type="xs:string" use="optional"/> <!-- iface
задавать только как iface_id! -->
        <xs:attribute name="prop" type="xs:string" use="optional"/>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out-->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="domain" type="domainType"/>
            <xs:element name="ifaces" type="ifacesType"/>
            <xs:element name="addresses" type="addressesType"/>
            <xs:element name="aliases" type="aliasesType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

aliases_list.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <!-- aliasType-->
    <xs:complexType name="aliasType">
        <xs:attribute name="domain" type="xs:string" use="required"/>
        <xs:attribute name="interface" type="xs:string" use="required"/>
        <xs:attribute name="interface_type" type="xs:string" use="optional"/>
        <xs:attribute name="interface_group" type="xs:string" use="optional"/>
        <xs:attribute name="address" type="xs:string" use="required"/>
        <xs:attribute name="iface_name" type="xs:string" use="optional"/>
        <xs:attribute name="active" type="xs:boolean" use="required"/>
        <xs:attribute name="binded" type="xs:boolean" use="optional"/>
        <xs:attribute name="displayName" type="xs:string" use="optional"/>
    </xs:complexType>
    <!-- errorType-->
    <xs:complexType name="errorType">
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute name="interface" type="xs:string"/>
                <xs:attribute name="address" type="xs:string"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:schema>

```

hc_aliases_list

Команда возвращает список алиасов, определенных в домене для конкретного адреса.

Аналог команды в CoCon:

```
domain/<DOMAIN>/alias/list
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа hc_aliases_list.xsd:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- aliasType-->
  <xs:complexType name="aliasType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="interface" type="xs:string" use="required"/>
    <xs:attribute name="interface_type" type="xs:string" use="optional"/>
    <xs:attribute name="interface_group" type="xs:string" use="optional"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="iface_name" type="xs:string" use="optional"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
    <xs:attribute name="binded" type="xs:boolean" use="optional"/>
    <xs:attribute name="displayName" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- errorType-->
  <xs:complexType name="errorType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="interface" type="xs:string"/>
        <xs:attribute name="address" type="xs:string"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:schema>
```

Пример

Запрос: http://192.168.1.21:9999/commands/alias_list

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_alias_list.xsd">
  <aliases domain="biysk.local" addr="24046{4-5}" />
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_alias_list.xsd">
  <aliases>
    <alias domain="biysk.local" interface="0616e1cdf89ba684" interface_type="sip"
interface_group="loc.gr" address="240464" active="true"/>
    <alias domain="biysk.local" interface="0616e1cdfa62ad0e" interface_type="sip"
interface_group="loc.gr" address="240465" active="true"/>
  </aliases>
</out>
```

hc_aliases_list2

Команда возвращает список алиасов, определенных в домене с фильтрацией по свойствам алиаса. В случае, если запрос выполняется длительное время (более 3х секунд), то команда прерывается, и возвращаются только те алиасы, что найдены на текущий момент. При этом возвращается курсор для продолжения поиска. Если же курсор не вернулся, это значит что все данные были возвращены.

Аналог команды в CoCon:

Нет прямого аналога. Работает как комбинация следующих команд:

```
domain/<DOMAIN>/alias/list
```

```
domain/<DOMAIN>/alias/info
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_list2

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- opType -->
  <xs:simpleType name="opType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="="/>
      <xs:enumeration value="~"/>
      <xs:enumeration value="/="/>
      <xs:enumeration value=">"/>
      <xs:enumeration value="<"/>
      <xs:enumeration value="=<"/>
      <xs:enumeration value=">="/>
      <xs:enumeration value="in"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- filterType -->
  <xs:complexType name="filterType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="op" type="opType" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- andFiltersType -->
  <xs:complexType name="andFiltersType">
    <xs:sequence>
      <xs:element name="filter" type="filterType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- filtersType -->
  <xs:complexType name="filtersType">
    <xs:sequence>
      <xs:element name="and" type="andFiltersType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- aliasType -->
  <xs:complexType name="aliasType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="iface" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
    <xs:attribute name="binded" type="xs:boolean" use="required"/>
    <xs:attribute name="display_name" type="xs:string" use="required"/>
    <xs:attribute name="regime" type="xs:string" use="required"/>
    <xs:attribute name="interface_name" type="xs:string" use="required"/>
    <xs:attribute name="interface_type" type="xs:string" use="required"/>
    <xs:attribute name="interface_owner" type="xs:string" use="required"/>
    <xs:attribute name="interface_group" type="xs:string" use="required"/>
  </xs:complexType>

```

```

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="filters" type="filtersType"/>
          </xs:all>
          <xs:attribute name="domain" type="xs:string" use="required"/>
          <xs:attribute name="limit" type="xs:string" use="required"/>
          <xs:attribute name="address_range" type="xs:string" use="optional"/>
          <xs:attribute name="cursor" type="xs:string" use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="result">
        <xs:complexType>
          <xs:sequence minOccurs="0" maxOccurs="unbounded">
            <xs:element name="alias" type="aliasType" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="cursor" type="xs:string" use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Запрос: http://192.168.1.21:9999/commands/alias_list2

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_alias_
_list2.xsd">
  <request address_range="240{500-510}" domain="biysk.local" limit="5">
    <filters>
      <and>
        <filter name="regime" op="==" value="accsusp"/>
      </and>
      <and>
        <filter name="active" op="==" value="true"/>
      </and>
    </filters>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_alias_l
ist2.xsd">
  <result>
    <alias domain="biysk.local" iface="0616e1cdf6d17ece" address="240462" active="true"
binded="true" display_name="" regime="accsusp" interface_name="240462@biysk.local" interface_type="
sip" interface_owner="sip1" interface_group="loc.gr"/>
    <alias domain="biysk.local" iface="0616e1ce554bc90e" address="240466" active="true"
binded="false" display_name="" regime="accsusp" interface_name="240466@biysk.local"
interface_type="sip" interface_owner="sip1" interface_group="loc.gr"/>
  </result>
</out>

```

Hc_alias - Управление алиасами

- [Hc_alias_list](#)
- [Hc_alias_list2](#)
- [Hc_alias_where](#)
- [Hc_alias_who](#)
- [Hc_alias_info](#)
- [Hc_alias_set](#)
- [Hc_alias_declare](#)
- [Hc_alias_remove](#)
- [Hc_alias_clean](#)

hc_alias_list

Команда возвращает список алиасов, определенных в домене и определенных в домене для определенного адреса или диапазона адресов.

Аналог команды в CoCon:

`domain/<DOMAIN>/alias/list`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/alias_list`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом hc_alias_list.xsd:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="aliases_list.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="aliases" minOccurs="1">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="addr" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="aliases" minOccurs="1">
          <xs:complexType>
            <xs:sequence minOccurs="0" maxOccurs="unbounded">
              <xs:element name="alias" type="aliasType" minOccurs="0"
maxOccurs="unbounded"/>
              <xs:element name="error" type="errorType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

aliases_list.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- aliasType-->
  <xs:complexType name="aliasType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="interface" type="xs:string" use="required"/>
    <xs:attribute name="interface_type" type="xs:string" use="optional"/>
    <xs:attribute name="interface_group" type="xs:string" use="optional"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="iface_name" type="xs:string" use="optional"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
    <xs:attribute name="binded" type="xs:boolean" use="optional"/>
    <xs:attribute name="displayName" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- errorType-->
  <xs:complexType name="errorType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="interface" type="xs:string"/>
        <xs:attribute name="address" type="xs:string"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/alias_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_alias_list.xsd">
  <aliases domain="biysk.local" addr="24046{2-4}" />
</in>

```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alias_list.xsd">
  <aliases>
    <alias domain="biysk.local" interface="0616e1cdf6d17ece" interface_type="sip"
  interface_group="loc.gr" address="240462" active="true"/>
    <alias domain="biysk.local" interface="0616e1cdf89ba684" interface_type="sip"
  interface_group="loc.gr" address="240464" active="true"/>
  </aliases>
</out>
```

hc_alias_list2

Команда возвращает список алиасов, определенных в домене с фильтрацией по свойствам алиаса.

Аналога команды в CoSop нет, является сочетанием двух команд:

domain/<DOMAIN>/alias/list

domain/<DOMAIN>/alias/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_list2

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом alias_list2.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- opType-->
  <xs:simpleType name="opType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="="/>
      <xs:enumeration value="~"/>
      <xs:enumeration value="/="/>
      <xs:enumeration value=">"/>
      <xs:enumeration value="<"/>
      <xs:enumeration value="=<"/>
      <xs:enumeration value=">="/>
      <xs:enumeration value="in"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- filterType -->
  <xs:complexType name="filterType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="op" type="opType" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- andFiltersType -->
  <xs:complexType name="andFiltersType">
    <xs:sequence>
      <xs:element name="filter" type="filterType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- filtersType -->
  <xs:complexType name="filtersType">
    <xs:sequence>
      <xs:element name="and" type="andFiltersType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- aliasType -->
  <xs:complexType name="aliasType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="iface" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
    <xs:attribute name="binded" type="xs:boolean" use="required"/>
    <xs:attribute name="display_name" type="xs:string" use="required"/>
    <xs:attribute name="regime" type="xs:string" use="required"/>
    <xs:attribute name="interface_name" type="xs:string" use="required"/>
    <xs:attribute name="interface_type" type="xs:string" use="required"/>
    <xs:attribute name="interface_owner" type="xs:string" use="required"/>
    <xs:attribute name="interface_group" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->

```

```

<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="filters" type="filtersType"/>
          </xs:all>
          <xs:attribute name="domain" type="xs:string" use="required"/>
          <xs:attribute name="limit" type="xs:string" use="required"/>
          <xs:attribute name="address_range" type="xs:string" use="optional"/>
          <xs:attribute name="cursor" type="xs:string" use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="result">
        <xs:complexType>
          <xs:sequence minOccurs="0" maxOccurs="unbounded">
            <xs:element name="alias" type="aliasType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="cursor" type="xs:string" use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/alias_list2:

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_alias_list2.xsd">
  <request address_range="240{500-510}" domain="biysk.local" limit="5">
    <filters>
      <and>
        <filter name="regime" op="==" value="accsusp"/>
      </and>
      <and>
        <filter name="active" op="==" value="true"/>
      </and>
    </filters>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alias_list2.xsd">
  <result>
    <alias domain="biysk.local" iface="061d4d840b68f853" address="240500" active="true"
    binded="true" display_name="" regime="accsusp" interface_name="240500@biysk.local"
    interface_type="sip" interface_owner="sip1" interface_group="loc.gr"/>
  </result>
</out>

```

hc_alias_where

Команда возвращает список задекларированных алиасов с определенным адресом (во всех доменах, со всеми интерфейсами).

Аналог команды в CoCon:

domain/<DOMAIN>/alias/where

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_where

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом alias_where.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
<xs:include schemaLocation="aliases_list.xsd"/>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="aliases" minOccurs="1">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="addr" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="aliases" minOccurs="1">
          <xs:complexType>
            <xs:sequence minOccurs="0" maxOccurs="unbounded">
              <xs:element name="alias" type="aliasType" minOccurs="0"
maxOccurs="unbounded"/>
              <xs:element name="error" type="errorType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

aliases_list.xsd:

Схема файла определена в разделе **hc_aliases_list**.

Пример:

Запрос: http://192.168.1.21:9999/commands/alias_where

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_alias_where.xsd">
  <aliases addr="240{4,5,6}0{0,1,3}" domain="biysk.local" />
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alias_where.xsd">
  <aliases>
    <error address="240400">{not_exists,"240400"}</error>
    <error address="240401">{not_exists,"240401"}</error>
    <error address="240403">{not_exists,"240403"}</error>
    <alias domain="biysk.local" interface="0616e1ce0437ed8d" address="240500"
iface_name="240500@biysk.local" active="true" binded="true" displayName=""/>
    <alias domain="biysk.local" interface="0616e1ce0437f792" address="240501"
iface_name="240501@biysk.local" active="true" binded="true" displayName=""/>
    <alias domain="biysk.local" interface="0616e1ce04381f14" address="240503"
iface_name="240503@biysk.local" active="true" binded="true" displayName=""/>
    <alias domain="biysk.local" interface="0616e1ce13af0be9" address="240600"
iface_name="240600@biysk.local" active="true" binded="true" displayName=""/>
    <alias domain="biysk.local" interface="0616e1ce13af16da" address="240601"
iface_name="240601@biysk.local" active="true" binded="true" displayName=""/>
    <alias domain="biysk.local" interface="0616e1ce13af2200" address="240603"
iface_name="240603@biysk.local" active="true" binded="true" displayName=""/>
  </aliases>
</out>
```

hc_alias_who

Команда возвращает список задекларированных алиасов с определенным интерфейсом (во всех доменах, со всеми адресами).

Аналог команды в CoCon:

domain/**<DOMAIN>**/**alias/who**

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_who

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом hc_alias_who.xsd:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="aliases_list.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="aliases" minOccurs="1">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="iface" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="aliases" minOccurs="1">
          <xs:complexType>
            <xs:sequence minOccurs="0" maxOccurs="unbounded">
              <xs:element name="alias" type="aliasType" minOccurs="0"
maxOccurs="unbounded"/>
              <xs:element name="error" type="errorType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Схема файла определена в разделе **hc_aliases_list**.

Пример:

Запрос: http://192.168.1.21:9999/commands/alias_who

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_alias_who.xsd">
  <aliases iface="e389d19152a30983" domain="biysk.local" />
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alias_who.xsd">
  <aliases>
    <alias domain="biysk.local" interface="0616e1ce13af2200" address="240603" active="true"
    binded="true" displayName="" />
  </aliases>
</out>
```

hc_alias_info

Команда возвращает информацию о свойствах алиасов. Кроме того, предусмотрена фильтрация по доменам, адресам, интерфейсам и свойствам алиаса посредством POST-запроса.

Аналог команды в CoCon:

`domain/<DOMAIN>/alias/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/alias_info`

- Для возврата свойства на уровне домена: указать `domain`
- Для возврата свойства на уровне адреса: указать `addr`
- Для возврата свойства на уровне домен+адрес: указать `domain`, `addr`
- Для возврата свойства на уровне домен+интерфейс: указать `domain`, `iface`
- Для возврата свойства на уровне домен+интерфейс+адрес: указать `domain`, `iface`, `addr`
- Для возврата свойства с именем `propertyName` уровня домен+интерфейс+адрес: указать `domain`, `iface`, `addr`, `prop`

Каждый из параметров «iface», «addr», «prop» может как присутствовать, так и отсутствовать. «propertyName» представляет собой имя свойства, значения которого необходимо получить. «iface_pattern» и «addr_pattern» представляют собой шаблон имени интерфейса и шаблон адреса соответственно. В шаблоне для отображения диапазона адресов/интерфейсов можно воспользоваться шаблоном “{from-to}”, где `from` и `to` – нижняя и верхняя граница

перечисления адресов/интерфейсов. Например: шаблон для адреса "550{02-10}01" выведет следующие адреса:

5500201, 5500301, 5500401, 5500501, 5500601, 5500701, 5500801, 5500901, 5501001 .

Для имен интерфейсов можно также воспользоваться данным шаблоном. Когда одновременно присутствуют параметры «iface» и «addr», каждый из них может принимать значение прочерк (_). Прочерк означает, что значение данного параметра вычисляется на основе второго.

В зависимости от того, какие параметры присутствуют в POST-запросе, возвращаются свойства, расположенные на разных уровнях.

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом aliases.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- propertyTypeType-->
  <xs:simpleType name="propertyTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="global"/>
      <xs:enumeration value="domain"/>
      <xs:enumeration value="domain_interface"/>
      <xs:enumeration value="domain_address"/>
      <xs:enumeration value="address"/>
      <xs:enumeration value="profile"/>
      <xs:enumeration value="alias"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="propertyTypeType" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          For other type of properties value will be formatted specifically.
          Key:   timetables\TIMETABLE_NAME
          Value: {"name":"TIMETABLE_NAME", "filters":[{"date":"DAYS_LIST_OR_DAY_TYPE_N
AME","from":"FROM_HH:FROM_MM","to":"TO_HH:TO_MM"}],"properties":{"description | KEY":"VALUE"}}
          Sample: {"name":"tt2","filters":[{"date":"1 2 3 4 5","from":"9:0","to":"12:59
"}, {"date":"HOLIDAY","from":"13:0","to":"17:59"}],"properties":{"description":"Мой 2","key3":{"he
llo,world"}}}
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- aliasType-->
  <xs:complexType name="aliasType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="iface" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- addressType-->
  <xs:complexType name="addressType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- ifaceType-->
  <xs:complexType name="ifaceType">

```

```

    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="iface" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- Properties elements -->
  <xs:complexType name="ifacesType">
    <xs:sequence>
      <xs:element name="iface" type="ifaceType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="addressesType">
    <xs:sequence>
      <xs:element name="address" type="addressType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="aliasesType">
    <xs:sequence>
      <xs:element name="alias" type="aliasType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="domainType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="aliases" minOccurs="1" maxOccurs="unbounded">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="addr" type="xs:string" use="optional"/>
            <xs:attribute name="iface" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- iface задавать только как iface_id! -->

```

```

        <xs:attribute name="prop" type="xs:string" use="optional"/>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="domain" type="domainType"/>
            <xs:element name="ifaces" type="ifacesType"/>
            <xs:element name="addresses" type="addressesType"/>
            <xs:element name="aliases" type="aliasesType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/alias_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="aliases.xsd">
  <aliases addr="24046{2-5}" domain="biysk.local" iface="_" />
</in>

```

Ответ:

200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="aliases.xsd">
  <aliases domain="biysk.local">
    <alias iface="0616e1cdf6d17ece" address="240462">
      <property name="user_agent" type="alias" value="Brcm-Callctrl/v1.10.3 M5T SIP Stack/
4.1.2.2"/>
      <property name="uid" type="alias" value="<<"0616e1cdf6fd6a2a">>"/>
      <property name="timezone" type="global" value="UTC+07:00"/>
      <property name="terminal_type" type="alias" value="smart"/>
      <property name="subscriber_portal\password" type="alias" value="9nMz5rjL"/>
      <property name="subscriber_portal\login" type="alias" value="240462"/>
      <property name="ss\mgm\telephone" type="global" value="enabled"/>
      <property name="screening" type="profile" value="networkProvided"/>
      <property name="profile" type="alias" value="user_default"/>
      <property name="pin" type="global" value="1111"/>
      <property name="npi" type="profile" value="isdntelephony"/>
      <property name="ni" type="profile" value="private"/>
      <property name="nai" type="profile" value="subscriberNumber"/>
      <property name="media-profile" type="domain" value="default"/>
      <property name="isIfaceActive" type="alias" value="true"/>
      <property name="isActive" type="alias" value="true"/>
      <property name="dtmf_source" type="alias" value="rfc+info"/>
      <property name="cdr_group" type="domain" value="bsk3"/>
      <property name="category" type="profile" value="10"/>
      <property name="access_type" type="domain" value="access_type0"/>
      <property name="access_group" type="profile" value="all"/>
    </alias>
    <alias iface="0616e1cdf89ba684" address="240464">
      <property name="user_agent" type="alias" value="TAU-2M.IP/2.0.0.229 SN/VI39005419
sofia-sip/1.12.10"/>
      <property name="uid" type="alias" value="<<"0616e1cdf8cda96a">>"/>
      <property name="timezone" type="global" value="UTC+07:00"/>
      <property name="terminal_type" type="alias" value="smart"/>
      <property name="subscriber_portal\password" type="alias" value="8I4GzvCi"/>
      <property name="subscriber_portal\login" type="alias" value="240464"/>
      <property name="ss\mgm\telephone" type="global" value="enabled"/>
      <property name="screening" type="profile" value="networkProvided"/>
      <property name="profile" type="alias" value="user_default"/>
      <property name="pin" type="global" value="1111"/>
      <property name="npi" type="profile" value="isdntelephony"/>
      <property name="ni" type="profile" value="private"/>
      <property name="nai" type="profile" value="subscriberNumber"/>
      <property name="media-profile" type="domain" value="default"/>
      <property name="last_incoming_call_info" type="alias"
value="{rtop_last_incoming_call_info,1,"240462",3559082181779416120, {1559,30929,916928}}"/>
      <property name="isIfaceActive" type="alias" value="true"/>
      <property name="isActive" type="alias" value="true"/>
    </alias>
  </aliases>
</out>

```

```

    <property name="dtmf_source" type="alias" value="rfc+info"/>
    <property name="cdr_group" type="domain" value="bsk3"/>
    <property name="category" type="profile" value="10"/>
    <property name="access_type" type="domain" value="access_type0"/>
    <property name="access_group" type="profile" value="all"/>
  </alias>
  <alias iface="0616e1cdfa62ad0e" address="240465">
    <property name="user_agent" type="alias" value="TAU-2M.IP/2.0.0.229 SN/VI39005419
sofia-sip/1.12.10"/>
    <property name="uid" type="alias" value="<<"0616e1cdfaa175a8">>"/>
    <property name="timezone" type="global" value="UTC+07:00"/>
    <property name="terminal_type" type="alias" value="smart"/>
    <property name="subscriber_portal\password" type="alias" value="9vFyIB2a"/>
    <property name="subscriber_portal\login" type="alias" value="240465"/>
    <property name="ss\mgm\telephone" type="global" value="enabled"/>
    <property name="screening" type="profile" value="networkProvided"/>
    <property name="profile" type="alias" value="user_default"/>
    <property name="pin" type="global" value="1111"/>
    <property name="npi" type="profile" value="isdntelephony"/>
    <property name="ni" type="profile" value="private"/>
    <property name="nai" type="profile" value="subscriberNumber"/>
    <property name="media-profile" type="domain" value="default"/>
    <property name="last_incoming_call_info" type="alias"
value="{rtop_last_incoming_call_info,1,"240474",4063428173976658745, {1558,932168,864198}}"/>
    <property name="isIfaceActive" type="alias" value="true"/>
    <property name="isActive" type="alias" value="true"/>
    <property name="dtmf_source" type="alias" value="rfc+info"/>
    <property name="cdr_group" type="domain" value="bsk3"/>
    <property name="category" type="profile" value="10"/>
    <property name="access_type" type="domain" value="access_type0"/>
    <property name="access_group" type="profile" value="all"/>
  </alias>
</aliases>
</out>

```

Ис_алиас_сет

Команда предназначена для задания свойств алиаса на различных уровнях. Свойства могут быть определены на уровнях: глобальный, домен, адрес, домен + адрес, домен + интерфейс, домен + адрес + интерфейс.

Аналог команды в CoSoc:

domain/<DOMAIN>/alias/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_set

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом `set_alias_properties.xsd`:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          For other type of properties value will be formatted specifically.
          Key:   timetables\TIMETABLE_NAME
          Value: {"name":"TIMETABLE_NAME", "filters":[{"date":"DAYS_LIST_OR_DAY_TYPE_NAME", "from":"FROM_HH:FROM_MM", "to":"TO_HH:TO_MM"}], "properties":{"description | KEY":"VALUE"}}
          Sample: {"name":"tt2", "filters":[{"date":"1 2 3 4 5", "from":"9:0", "to":"12:59"}, {"date":"HOLIDAY", "from":"13:0", "to":"17:59"}], "properties":{"description":"Мой 2", "key3":{"hello, world}}}}
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="aliases">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="addr" type="xs:string" use="optional"/>
            <xs:attribute name="iface" type="xs:string" use="optional">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  iface :: IfacePattern1;IfacePattern2;...IfacePatternN
                  IfacePatternN :: {From-To}@domain
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="error">

```

```

        <xs:complexType>
          <xs:attribute name="domain" type="xs:string" use="optional" />
          <xs:attribute name="iface" type="xs:string" use="optional" />
          <xs:attribute name="addr" type="xs:string" use="optional" />
          <xs:attribute name="reason" type="xs:string" use="required" />
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Установить режим для диапазона алиасов и всех принадлежащих им интерфейсов

Запрос: http://192.168.1.21:9999/commands/alias_set

Параметры:

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="set_alias_properties.xsd">
  <aliases addr="24046{2,6}" iface="_" domain="biysk.local">
    <property name="regime" value="accsusp"/>
  </aliases>
</in>

```

Ответ: 201

[Hc_alias_declare](#)

Команда предназначена для создания алиасов в домене.

Аналог команды в CoCon:

domain/<DOMAIN>/alias/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_declare

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом hc_alias_declare.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />

  <!-- causeType-->
  <xs:simpleType name="causeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="already_exists"/>
      <xs:enumeration value="licence_limit"/>
      <xs:enumeration value="domain_limit"/>
      <xs:enumeration value="no_such_iface"/>
      <xs:enumeration value="bad_owner"/>
      <xs:enumeration value="bad_group"/>
      <xs:enumeration value="cannot_get_iface"/>
      <xs:enumeration value="invalid_address_format"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- warningType-->
  <xs:complexType name="warningType">
    <xs:attribute name="cause" type="causeType"/>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="interface" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- aliasType-->
  <xs:complexType name="aliasType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="interface" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
  </xs:complexType>

  <!-- successType-->
  <xs:complexType name="successType">
    <xs:sequence>
      <xs:element name="alias" type="aliasType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- warningsType-->
  <xs:complexType name="warningsType">
    <xs:sequence>
      <xs:element name="warning" type="warningType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

```

```

<!-- resultType-->
<xs:complexType name="resultType">
  <xs:sequence>
    <xs:element name="success" type="successType"/>
    <xs:element name="warnings" type="warningsType"/>
  </xs:sequence>
</xs:complexType>

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:attribute name="domain" type="xs:string" use="required"/>
          <xs:attribute name="address" type="xs:string" use="required"/>
          <xs:attribute name="iface" type="xs:string" use="required"/>
          <xs:attribute name="active" type="xs:boolean" use="required"/>
          <xs:attribute name="owner" type="xs:string" use="optional"/>
          <xs:attribute name="group" type="xs:string" use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType" />
      <xs:element name="error" type="errorType" />
    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/alias_declare

```
<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_alias_declare.xsd">
  <request domain="biysk.local" address="600" iface="0616e1ce554bc90e" active="true"
  owner="sip1" group="loc.gr"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_alias_declare.xsd">
  <result>
    <success>
      <alias domain="biysk.local" interface="0616e1ce554bc90e" address="600" active="true"/
    >
    </success>
    <warnings/>
  </result>
</out>
```

[Hc_alias_remove](#)

Команда предназначена для удаления алиасов из домена.

Аналог команды в CoCon:

domain/<DOMAIN>/alias/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_remove

XSD-схема XML-файла с ответом hc_alias_remove.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <!-- aliasType-->
  <xs:complexType name="aliasType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="iface" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- aliasesType-->
  <xs:complexType name="aliasesType">
    <xs:sequence>
      <xs:element name="alias" type="aliasType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="aliases" type="aliasesType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

Пустой XML

Пример:

Запрос: http://192.168.23.34:9999/commands/alias_remove

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_alias_remove.xsd">
  <request>
    <aliases>
      <alias domain="biysk.local" address="600" iface="0616e1ce554bc90e"/>
    </aliases>
  </request>
</in>
```

Ответ: 204.

[Hc_alias_clean](#)

Команда предназначена для удаления свойства алиаса домена.

Аналог команды в CoCon:

domain/<DOMAIN>/alias/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/alias_clean

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом aliases.xsd:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- propertyTypeType-->
  <xs:simpleType name="propertyTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="global"/>
      <xs:enumeration value="domain"/>
      <xs:enumeration value="domain_interface"/>
      <xs:enumeration value="domain_address"/>
      <xs:enumeration value="address"/>
      <xs:enumeration value="profile"/>
      <xs:enumeration value="alias"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="propertyTypeType" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          For other type of properties value will be formatted specifically.
          Key:   timetables\TIMETABLE_NAME
          Value: {"name":"TIMETABLE_NAME", "filters":[{"date":"DAYS_LIST_OR_DAY_TYPE_N
AME","from":"FROM_HH:FROM_MM","to":"TO_HH:TO_MM"}],"properties":{"description | KEY":"VALUE"}}
          Sample: {"name":"tt2","filters":[{"date":"1 2 3 4 5","from":"9:0","to":"12:59
"}, {"date":"HOLIDAY","from":"13:0","to":"17:59"}],"properties":{"description":"Мой 2","key3":{"he
llo,world"}}}
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- aliasType-->
  <xs:complexType name="aliasType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="iface" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- addressType-->
  <xs:complexType name="addressType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- ifaceType-->
  <xs:complexType name="ifaceType">

```

```

    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- errorType-->
  <xs:complexType name="errorType">
    <xs:attribute name="iface" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- Properties elements-->
  <xs:complexType name="ifacesType">
    <xs:sequence>
      <xs:element name="iface" type="ifaceType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="addressesType">
    <xs:sequence>
      <xs:element name="address" type="addressType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="aliasesType">
    <xs:sequence>
      <xs:element name="alias" type="aliasType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="domainType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="aliases" minOccurs="1" maxOccurs="unbounded">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="addr" type="xs:string" use="optional"/>
            <xs:attribute name="iface" type="xs:string" use="optional"/> <!-- iface
задавать только как iface_id! -->
            <xs:attribute name="prop" type="xs:string" use="optional"/>

```

```

        </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="domain" type="domainType"/>
            <xs:element name="ifaces" type="ifacesType"/>
            <xs:element name="addresses" type="addressesType"/>
            <xs:element name="aliases" type="aliasesType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/alias_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="aliases.xsd">
    <aliases addr="24052{0-1}" iface="_" domain="biysk.local" prop="regime" />
</in>

```

Ответ:

204

Нс_aon_category_options - Команды мапинга категории АОН в категорию ОКС

- [aon_category_options_clean](#)
- [aon_category_options_info](#)
- [aon_category_options_set](#)

[aon_category_options_clean](#)

Команда для сброса в значение по умолчанию таблицы мапинга категории АОН в категорию ОКС (для определенного АОН)

Аналог команды в CoCon:

domain/properties/aon/clean

Метод http запроса

POST

Шаблон http запроса

http://host:port/commands/aon_category_options_clean

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема xml файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean AON category map.
      Usage: http://server:9999/commands/aon_category_options_clean.
      Implementation: hc_aon_category_options_clean.erl.
      Copyright (c) 2019, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema

```

Пример 1

Запрос: http://192.168.1.21:9999/commands/aon_category_options_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="aon_category_options_clean.xsd">
  <request>
    <domain_options>
      <system>
        <option name="aon_0" />
      </system>
      <domain name="biysk.local">
        <option name="aon_0" />
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="aon_category_options_clean.xsd">
  <result>
    <domain_options>
      <system>
        <success option="aon_0"/>
      </system>
      <domain name="biysk.local">
        <success option="aon_0"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Пример 2

Запрос: http://192.168.1.21:9999/commands/aon_category_options_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="aon_category_options_clean.xsd">
  <request>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <option name="aon_0" />
        <option name="aon_2" />
        <option name="aon_3" />
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="aon_category_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="aon_0"/>
        <success option="aon_2"/>
        <success option="aon_3"/>
      </domain>
    </domain_options>
  </result>
</out>

```

[aon_category_options_info](#)

Команда для просмотра таблицы перевода категории АОН в категорию ОКС, как на всей системе целиком, так и на определенном домене.

Аналог команды в CoCon:

domain/properties/aon/info

Метод http запроса

POST

Шаблон http запроса

http://host:port/commands/aon_category_options_info

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема xml файла с ответом:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show AON to Category map.
      Usage: http://server:9999/commands/aon\_category\_options\_info.
      Implementation: hc_aon_category_options_info.erl.
      Copyright (c) 2019, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- Types -->
  <xs:simpleType name="aonCategoryMapType">
    <xs:annotation>
      <xs:documentation xml:lang="en">
        The list of categories joined by comma
        Example: 224, 225
      </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string"/>
  </xs:simpleType>

  <xs:complexType name="aonCategoryOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="aon_0" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_1" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_2" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_3" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_4" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_5" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_6" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_7" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_8" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_9" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_10" type="aonCategoryMapType" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>

```

```

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1

Запрос: http://192.168.1.21:9999/commands/aon_category_options_info

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="aon_category_options_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="aon_category_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="aonCategoryOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="aonCategoryOptionsType" aon_9="229" aon_8="12" aon_7="227,240"
aon_6="15" aon_5="226" aon_4="11" aon_3="228" aon_2="225" aon_10="" aon_1="10" aon_0=""/>
      </domain>
    </domain_options>
  </result>
</out>

```

Пример 2

Запрос: http://192.168.1.21:9999/commands/aon_category_options_info

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="aon_category_options_info.xsd">
  <request>
    <domain_options/>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="aon_category_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="aonCategoryOptionsType" aon_9="229" aon_8="12" aon_7="227,240"
aon_6="15" aon_5="226" aon_4="11" aon_3="228" aon_2="225" aon_10="" aon_1="10" aon_0=""/>
      </system>
    </domain_options>
  </result>
</out>

```

`aon_category_options_set`

Команда для редактирования таблицы мапинга категории АОН в категорию ОКС, как на всей системе целиком, так и на определенном домене.

Аналог команды в CoCon:

`domain/properties/aon/set`

Метод http запроса

POST

Шаблон http запроса

http://host:port/commands/aon_category_options_set

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема xml файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set domain properties.
      Usage: http://server:9999/commands/domain_properties_set.
      Implementation: hc_domain_properties_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- Types -->
  <xs:simpleType name="aonCategoryMapType">
    <xs:annotation>
      <xs:documentation xml:lang="en">
        The list of categories joined by comma
        Example: 224, 225
      </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
  <!-- aonCategoryOptionsType -->
  <xs:complexType name="aonCategoryOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="aon_0" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_1" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_2" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_3" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_4" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_5" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_6" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_7" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_8" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_9" type="aonCategoryMapType" use="optional"/>
        <xs:attribute name="aon_10" type="aonCategoryMapType" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->

```

```

<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/aon_category_options_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="aon_category_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="aonCategoryOptionsType" aon_0="1,2" />
      </system>
      <domain name="biysk.local">
        <options xs:type="aonCategoryOptionsType" aon_0="3, 4" aon_5="226,227" />
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="aon_category_options_set.xsd">
  <result>
    <domain_options>
      <system>
        <success option="aon_0"/>
      </system>
      <domain name="biysk.local">
        <success option="aon_0"/>
        <success option="aon_5"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_backup - Команды копирования и восстановления настроек

- [Нс_backup_restore_domain](#)
- [Нс_backup_restore](#)
- [Нс_backup_store_domain](#)
- [Нс_backup_store](#)

Нс_backup_restore_domain

Команда восстанавливает настройки домена (из формата base64).

Аналог команды в CoCon:

domain/<DOMAIN>/backup/restore

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/backup_restore_domain

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом `hc_backup_restore_domain.xsd`:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:choice>
        <xs:element name="backup">
          <xs:complexType>
            <xs:simpleContent>
              <xs:extension base="xs:string">
                <xs:attribute name="storage" type="xs:string" use="required"/>
                <xs:attribute name="domain" type="xs:string" use="required"/>
              </xs:extension>
            </xs:simpleContent>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/backup_restore_domain

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_backup_restore_domain.xsd">
  <backup storage="ds1@ecss1" domain="test.domain">g2gFZAAQcnRvcC1leHBvcnQtZGF
    ...
    /KQL0ZhvwlhpbicYDC6otx+mQ0Ecb8KguJSB3ueSMqZxABbh8uPvjf/wNflcoU</backup>
</in>

```

Ответ: 204

hc_backup_restore

Команда восстанавливает настройки системы (из формата base64).

Аналог команды в CoCon:

cluster/storage/<STORAGE>/backup/restore

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/backup_restore

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом hc_backup_restore.xsd:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:choice>
        <xs:element name="backup">
          <xs:complexType>
            <xs:simpleContent>
              <xs:extension base="xs:string">
                <xs:attribute name="storage" type="xs:string" use="required"/>
              </xs:extension>
            </xs:simpleContent>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: `http://192.168.1.21:9999/commands/backup_restore:`

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_backup_restore.xsd">
  <backup storage="ds1@ecss1">H4sIAAAAAAAAAA+
    ...
    Gwf9vQv+P0f8hL/0fAAAAEuii/wN5fQCUe7BCAHIRAA==</backup>
</in>
```

Ответ: 204

hc_backup_store_domain

Команда возвращает настройки домена (в формате base64).

Аналог команды в CoCon:

`domain/<DOMAIN>/backup/store`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/store_domain`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом hc_backup_store_domain.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:choice>
        <xs:element name="store">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:choice>
              <xs:element name="backup">
                <xs:complexType>
                  <xs:simpleContent>
                    <xs:extension base="xs:string">
                      <xs:attribute name="storage" type="xs:string"
use="required"/>
                      <xs:attribute name="domain" type="xs:string"
use="required"/>
                    </xs:extension>
                  </xs:simpleContent>
                </xs:complexType>
              </xs:element>
            </xs:choice>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/backup_store_domain

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_backup_store_domain.xsd">
    <store storage="ds1@ecss1" domain="test.domain"/>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="hc_backup_store_domain.xsd">
    <response>
        <backup storage="ds1@ecss1" domain="test.domain">g2gFZAAQcnRvcC1leHBvcnQtZGF
            ...
            /KQL0ZhvwlhpBicYDC6otx+mQ0Ecb8KguJSB3ueSMqZxABbh8uPvjf/wNf1coU</backup>
    </response>
</out>
```

hc_backup_store

Команда возвращает настройки системы (в формате base64).

Аналог команды в CoCon:

cluster/storage/<STORAGE>/backup/store

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/backup_store

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом hc_backup_store.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:choice>
        <xs:element name="store">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:choice>
              <xs:element name="backup">
                <xs:complexType>
                  <xs:simpleContent>
                    <xs:extension base="xs:string">
                      <xs:attribute name="storage" type="xs:string"
use="required"/>
                    </xs:extension>
                  </xs:simpleContent>
                </xs:complexType>
              </xs:element>
            </xs:choice>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/backup_store

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_backup_store.xsd">
  <store storage="ds1@ecss1"/>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_backup_store.xsd">
  <response>
  <backup storage="ds1@ecss1">H4sIAAAAAAAAAA+
  ...
  Gwf9vQv+P0f8hL/0fAAAAEuii/wN5fQCUe7BCAHIRAA==</backup>
  </response>
</out>
```

Hc_bridge - Управление бриджами

- [Hc_bridge_declare](#)
- [Hc_bridge_info](#)
- [Hc_bridge_remove](#)
- [Hc_bridge_cleanup](#)
- [Hc_bridge_change](#)

Hc_bridge_declare

Команда предназначена для декларации бриджа между двумя виртуальными АТС в рамках одного ECSS-10.

Аналог команды в CoCon:

/bridge/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/bridge_declare

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="bridge.xsd"/>
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="bridge" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- warnType -->
  <xs:complexType name="warnType">
    <xs:attribute name="bridge" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="bridge" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="bridge" type="bridgeType" minOccurs="0" maxOccurs="
unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok" type="okType" minOccurs="0" maxOccurs="unbounde
d"/>
              <xs:element name="warn" type="warnType" minOccurs="0" maxOccurs="unbo
unded"/>
              <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="un
bounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```



```

    </xs:complexType>
  </xs:element>
</xs:schema>

```

bridge.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- directionType -->
  <xs:complexType name="directionType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="interface" type="xs:string" use="required"/>
    <xs:attribute name="trunkgroupid" type="xs:string" use="required"/>
    <xs:attribute name="context" type="xs:string" use="required"/>
    <xs:attribute name="np" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- bridgeType -->
  <xs:complexType name="bridgeType">
    <xs:sequence>
      <xs:element name="a" type="directionType" />
      <xs:element name="b" type="directionType" />
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="strict" type="xs:boolean" use="optional"/>
    <xs:attribute name="bandwidth_in" type="xs:string"/> <!-- non-negative integer or
'unbounded' string -->
    <xs:attribute name="bandwidth_out" type="xs:string"/> <!-- non-negative integer or
'unbounded' string -->
    <xs:attribute name="bandwidth_total" type="xs:string"/> <!-- non-negative integer or
'unbounded' string -->
  </xs:complexType>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/bridge_declare

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="bridge_declare.xsd">
  <request>
    <bridge name="biysk.local_to_test.biysk" strict="false">
      <a domain="biysk.local" interface="bridge:p_to_a" trunkgroupid="tg:p_to_a" context="ctx_from_local"/>
      <b domain="test.biysk" interface="bridge:a_to_p" trunkgroupid="tg:a_to_p" context="ctx_from_local"/>
    </bridge>
  </request>
</in>
```

Ответ 200:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="bridge_declare.xsd">
  <response>
    <ok bridge="biysk.local_to_test.biysk"/>
  </response>
</out>
```

Ис_bridge_info

Команда предназначена для просмотра информации о всех (или определенном) бриджах, декларированных в системе.

Аналог команды в CoCon:

/bridge/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/bridge_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="bridge.xsd"/>

  <xs:complexType name="bridgeInType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="bridge" type="bridgeInType" minOccurs="0" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="bridge" type="bridgeType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/bridge_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="bridge_info.xsd">
  <bridge name="biysk.local_to_test.biysk" />
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="bridge_info.xsd">
  <bridge name="biysk.local_to_test.biysk" bandwidth_in="unbounded" bandwidth_out="unbounded"
bandwidth_total="unbounded" strict="false">
    <a domain="biysk.local" np="default" interface="bridge:p_to_a" trunkgroupid="tg:p_to_a"
context="ctx_from_local"/>
    <b domain="test.biysk" np="default" interface="bridge:a_to_p" trunkgroupid="tg:a_to_p"
context="ctx_from_local"/>
  </bridge>
</out>
```

Чтобы получить список всех бриджей нужно выполнить запрос:

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="bridge_info.xsd" />
```

Ис_bridge_remove

Команда предназначена для удаления ранее декларированного бриджа из системы.

Аналог команды в CoCon:

/bridge/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/bridge_remove

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="bridgeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="bridge" type="bridgeType" minOccurs="0" maxOccurs="
unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/bridge_remove

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="bridge_remove.xsd">
  <request>
    <bridge name="biysk.local_to_test.biysk1"/>
  </request>
</in>

```

Ответ: 200

```
Bridges successfully removed: biysk.local_to_test.biysk1
```

Ответ: 404

```

<?xml version="1.0" encoding="UTF-8"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="common_error.xsd">
  <error cmd="bridge_remove" reason="error" body="Bridges are undefined" />
</out>

```

Нс_bridge_cleanup

Команда предназначена для удаления всех, декларированных в системе мостов.

Аналог команды в CoCon:

/bridge/cleanup

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/bridge_cleanup

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="cleanup" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/bridge_cleanup

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="bridge_cleanup.xsd">
  <cleanup />
</in>
```

Ответ: 204

Нс_bridge_change

Команда предназначена для изменения свойств декларированных в системе мостов.

Аналог команды в CoCon:

/bridge/change

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/bridge_change

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="bridge.xsd"/>
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="bridge" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- warnType -->
  <xs:complexType name="warnType">
    <xs:attribute name="bridge" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="bridge" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="bridge" type="bridgeType" minOccurs="0" maxOccurs="
unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok" type="okType" minOccurs="0" maxOccurs="unbounde
d"/>
              <xs:element name="warn" type="warnType" minOccurs="0" maxOccurs="unbo
unded"/>
              <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="un
bounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```



```

    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/bridge_change

```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="bridge_change.xsd">
  <request>
    <bridge name="biysk.local_to_test.biysk">
      <a domain="biysk.local" interface="bridge:p_to_a" np="default" trunkgroupid="tg:p_to_a" context="ctx_from_local"/>
      <b domain="test.biysk" interface="bridge:a_to_p" np="default" trunkgroupid="tg:a_to_p" context="ctx_from_local_1"/>
    </bridge>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="bridge_change.xsd">
  <response>
    <ok bridge="biysk.local_to_test.biysk"/>
  </response>
</out>

```

Hc_calendar_timetable - Команды управления календарем и расписанием

- [Hc_calendar_timetable_show](#)
- [Hc_calendar_timetable_set](#)
- [Hc_calendar_timetable_remove](#)

Hc_calendar_timetable_show

Команда позволяет вывести календарь и действующее расписание.

Аналог команды в CoCon:

`/domain/<DOMAIN>/calendar/timetable/show`

Шаблон URL запроса

http://host:port/commands/calendar_timetable_show

Код HTTP-ответа:

200 – в случае успеха;

404 – в случае ошибки.

XML-схема

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>
  <!-- timetableType -->
  <xs:complexType name="timetableType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:element name="timetable" type="timetableType" minOccurs="0" maxOccurs="unbounded"
/>
    </xs:sequence>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="optional"/>
            <xs:attribute name="timetable" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType"/>
        <!--Positive case-->
        <xs:element name="result" type="resultType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример

Запрос:

http://192.168.1.21:9999/commands/calendar_timetable_show

```
<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_calendar_timetable_show.xsd">
  <request domain="biysk.local"/>
</in>
```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_calendar_timetable_show.xsd">
  <result>
    <timetable name="tt4" value="{\"name\":\"tt4\", \"filters\": [{\"date\": \"1 2 3 4 5\", \"from\": \"8:0\", \"to\": \"17:0\"}], \"properties\": []}\"/>
  </result>
</out>
```

hc_calendar_timetable_set

Команда предназначена для установки нового расписания, рабочих, выходных, праздничных дней.

Аналог команды в CoCon:

```
/domain/<DOMAIN>/calendar/timetable/add-interval
/domain/<DOMAIN>/calendar/timetable/remove-interval
```

Шаблон URL запроса

http://host:port/commands/calendar_timetable_set

Код HTTP-ответа:

200 – в случае успеха;

404 – в случае ошибки.

XML-схема

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<!-- <error> -->
  <xs:complexType name="errorType">
    <xs:attribute name="name" type="xs:string" use="optional" />
    <xs:attribute name="reason" type="xs:string" use="optional" />
    <!-- entity="[nonempty_string]" -->
  </xs:complexType>
  <!-- timetableType -->
  <xs:complexType name="timetableType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- requestType-->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="timetable" type="timetableType" minOccurs="0" maxOccurs="unbounded"
/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:choice>
      <xs:sequence>
        <xs:element name="error" type="errorType"/>
      </xs:sequence>
      <!--Positive case-->
      <xs:element name="ok"/>
    </xs:choice>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type = "requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="result" type="resultType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/calendar_timetable_set

```
<?xml version="1.0"?>
  <in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_calendar_timetable_set.xsd">
    <request domain="test.domain">
      <timetable name="tt4"
value="{&quot;name&quot;:&quot;tt4&quot;,&quot;filters&quot;:
[&quot;date&quot;:&quot;1 2 3 4 5&quot;,&quot;from&quot;:&quot;
9:0&quot;,&quot;to&quot;:&quot;18:0&quot;}],&quot;properties&quot;:[]}">
      </request>
    </in>
```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_calendar_timetable_set.xsd">
  <result>
    <ok/>
  </result>
</out>
```

[Hc_calendar_timetable_remove](#)

Команда предназначена для удаления текущего календаря и расписания работы.

Аналог команды в CoCon:

domain/<DOMAIN>/calendar/timetable/remove

Шаблон URL запроса

http://host:port/commands/calendar_timetable_remove

Код HTTP-ответа:

200 – в случае успеха;

404 – в случае ошибки.

XML-схема

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<!-- <error> -->
  <xs:complexType name="errorType">
    <xs:attribute name="name" type="xs:string" use="optional" />
    <xs:attribute name="reason" type="xs:string" use="optional" />
    <!-- entity="[nonempty_string]" -->
  </xs:complexType>
<!-- timetableType -->
  <xs:complexType name="timetableType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
<!-- requestType-->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="timetable" type="timetableType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="optional"/>
  </xs:complexType>
<!-- resultType-->
  <xs:complexType name="resultType">
    <xs:choice>
      <xs:sequence>
        <xs:element name="error" type="errorType"/>
      </xs:sequence>
      <!--Positive case-->
      <xs:element name="ok"/>
    </xs:choice>
  </xs:complexType>
<!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
<!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="result" type="resultType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/calendar_timetable_remove

```
<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_calendar_timetable_remove.xsd">
  <request domain="biysk.local">
    <timetable name="tt4"/>
  </request>
</in>
```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_calendar_timetable_remove.xsd">
  <result>
    <ok/>
  </result>
</out>
```

Hc_calls_active - Управление сервисом service actual calls

- [Hc_calls_active](#)

Hc_calls_active

Команда отображает список активных разговоров.

Аналог команды в CoCon:

```
/domain/<DOMAIN>/calls/list --active true
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/calls_active

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="calls_limit" type="xs:integer"/>
  </xs:complexType>

  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="callType">
<!--
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
-->
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="call_ref" type="xs:string"/>
    <xs:attribute name="stage" type="xs:string"/>
    <xs:attribute name="ocgpn_a" type="xs:string"/>
    <xs:attribute name="ocdpn_b" type="xs:string"/>
    <xs:attribute name="start_time" type="xs:string"/>
    <xs:attribute name="alerting_time" type="xs:string"/>
    <xs:attribute name="answer_time" type="xs:string"/>
    <xs:attribute name="display_name_a" type="xs:string"/>
    <xs:attribute name="display_name_b" type="xs:string"/>
    <xs:attribute name="category_a" type="xs:string"/>
    <xs:attribute name="category_b" type="xs:string"/>
    <xs:attribute name="iface_a" type="xs:string"/>
    <xs:attribute name="iface_b" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="domainOutType">
    <xs:sequence>
      <xs:element name="call" type="callType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>

```

```

    <xs:sequence>
      <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:all>
</xs:complexType>
</xs:element>

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <!--Positive case-->
      <xs:element name="response">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain" type="domainOutType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
      <!--Or error-->
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/calls_active

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_calls_a
ctive.xsd">
  <request>
    <domain name="biysk.local" calls_limit="10"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_calls_active.xsd">
  <response>
    <domain name="biysk.local">
      <call id="" call_ref="3033252287" stage="answered" iface_a="bsk1" iface_b="ems1"
display_name_a="241000" display_name_b="" start_time="02.10.2018 12:57:55" alerting_time="02.10.2018 12:57:55" answer_time="02.10.2018 12:57:56"/>
      <call id="" call_ref="3036947638" stage="answered" iface_a="bsk1" iface_b="ems1"
display_name_a="241001" display_name_b="" start_time="02.10.2018 12:57:56" alerting_time="02.10.2018 12:57:56" answer_time="02.10.2018 12:57:56"/>
      <call id="" call_ref="3037446783" stage="answered" iface_a="bsk1" iface_b="ems1"
display_name_a="241002" display_name_b="" start_time="02.10.2018 12:57:56" alerting_time="02.10.2018 12:57:56" answer_time="02.10.2018 12:57:57"/>
      <call id="" call_ref="3041141736" stage="answered" iface_a="bsk1" iface_b="ems1"
display_name_a="241003" display_name_b="" start_time="02.10.2018 12:57:57" alerting_time="02.10.2018 12:57:57" answer_time="02.10.2018 12:57:57"/>
      <call id="" call_ref="3041641576" stage="answered" iface_a="bsk1" iface_b="ems1"
display_name_a="241004" display_name_b="" start_time="02.10.2018 12:57:57" alerting_time="02.10.2018 12:57:57" answer_time="02.10.2018 12:57:58"/>
      <call id="" call_ref="3045335875" stage="answered" iface_a="bsk1" iface_b="ems1"
display_name_a="241005" display_name_b="" start_time="02.10.2018 12:57:58" alerting_time="02.10.2018 12:57:58" answer_time="02.10.2018 12:57:58"/>
      <call id="" call_ref="3045835816" stage="answered" iface_a="bsk1" iface_b="ems1"
display_name_a="241006" display_name_b="" start_time="02.10.2018 12:57:58" alerting_time="02.10.2018 12:57:58" answer_time="02.10.2018 12:57:59"/>
      <call id="" call_ref="3049530097" stage="answered" iface_a="bsk1" iface_b="ems1"
display_name_a="241007" display_name_b="" start_time="02.10.2018 12:57:59" alerting_time="02.10.2018 12:57:59" answer_time="02.10.2018 12:57:59"/>
      <call id="" call_ref="3050029889" stage="answered" iface_a="bsk1" iface_b="ems1"
display_name_a="241008" display_name_b="" start_time="02.10.2018 12:57:59" alerting_time="02.10.2018 12:57:59" answer_time="02.10.2018 12:58:00"/>
      <call id="" call_ref="3053723989" stage="answered" iface_a="bsk1" iface_b="ems1"
display_name_a="241009" display_name_b="" start_time="02.10.2018 12:58:00" alerting_time="02.10.2018 12:58:00" answer_time="02.10.2018 12:58:00"/>
    </domain>
  </response>
</out>

```

Hc_calls_history - Управление историей вызовов

- [Hc_calls_history](#)
- [Hc_purge_calls_db](#)

Нс_calls_history

Команда выдает историю вызовов, включая вызовы, активные в момент запроса.

Аналог команды в CoCon:

`/domain/<DOMAIN>/calls/list`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:port/commands/calls_history

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="hc_calls_history">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Shows calls history on SSW.
      Copyright (C) 2014, Eltex. All right reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- Types -->
  <xs:simpleType name="dateTimeType">
    <xs:restriction base="xs:string">
      <xs:pattern value="\d{2}\.\d{2}\.\d{4}\s+\d{2}:\d{2}:\d{2}"/>
      <!-- DD.MM.YYYY HH:MM:SS -->
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="addressType">
    <xs:restriction base="xs:string">
      <xs:pattern value="%( [0-9ABCD*#] )%?"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="addressListType">
    <xs:restriction base="xs:string">
      <xs:pattern value="%( [0-9ABCD*#] )%?, ?) *"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="faxFilterType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="success"/>
      <xs:enumeration value="failed"/>
      <xs:enumeration value="all"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="callsTypeType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="active"/>
      <xs:enumeration value="history"/>
      <xs:enumeration value="all"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="nonEmptyTokenType" use="required"/>
    <xs:attribute name="limit" type="xs:positiveInteger" default="50"/>
    <xs:attribute name="tag" type="xs:base64Binary" use="optional"/>
    <xs:attribute name="iface" type="xs:string"/>
    <xs:attribute name="addr" type="addressListType">
      <xs:annotation>
        <xs:documentation xml:lang="en">

```

The list of addresses split by comma. In **case** of addr prefix or suffix is % - it means any digits.

```

    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="pair" type="addressListType">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      The list of 2 addresses split by comma.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="addr_a" type="addressType"/>
<xs:attribute name="addr_b" type="addressType"/>
<xs:attribute name="from_date" type="dateTimeType"/>
<xs:attribute name="to_date" type="dateTimeType"/>
<xs:attribute name="fax_filter" type="faxFilterType">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show calls only with faxes:
      success - select calls only successfully received faxes;
      failed - select calls only unsuccessfully received faxes;
      all - select calls with any received faxes.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="calls_type" type="callsTypeType" default="all">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Selected calls' type:
      active - select only active calls;
      history - select only released calls;
      all - select active and released calls.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="call_id" type="nonEmptyTokenType" use="optional"/>
<xs:attribute name="mode" type="xs:string" use="optional"/>
<xs:attribute name="is_ss_present" type="xs:boolean" use="optional"/>
</xs:complexType>
<xs:complexType name="callType">
  <xs:attribute name="call_id" type="nonEmptyTokenType"/>
  <xs:attribute name="branch_id" type="xs:nonNegativeInteger"/>
  <xs:attribute name="parent_branch_id" type="xs:string"/>
  <xs:attribute name="call_ref" type="xs:integer"/>
  <xs:attribute name="stage" type="nonEmptyTokenType"/>
  <xs:attribute name="oaddr_a" type="addressType"/>
  <xs:attribute name="oaddr_b" type="addressType"/>
  <xs:attribute name="addr_a" type="addressType"/>
  <xs:attribute name="addr_b" type="addressType"/>

```

```

<xs:attribute name="ni_a" type="nonEmptyTokenType"/>
<xs:attribute name="ni_b" type="nonEmptyTokenType"/>
<xs:attribute name="release_initiator" type="nonEmptyTokenType"/>
<xs:attribute name="start_time" type="nonEmptyNormalizedStringType"/>
<xs:attribute name="alerting_time" type="xs:normalizedString"/>
<xs:attribute name="answer_time" type="xs:normalizedString"/>
<xs:attribute name="release_time" type="xs:normalizedString"/>
<xs:attribute name="duration_time" type="xs:integer"/>
<xs:attribute name="display_name_a" type="xs:normalizedString"/>
<xs:attribute name="display_name_b" type="xs:normalizedString"/>
<xs:attribute name="codec_a" type="xs:string" use="optional"/>
<xs:attribute name="codec_b" type="xs:string" use="optional"/>
<xs:attribute name="category_a" type="xs:token"/>
<xs:attribute name="category_b" type="xs:token"/>
<xs:attribute name="iface_a" type="xs:token"/>
<xs:attribute name="iface_b" type="xs:token"/>
<xs:attribute name="trunk_group_id_a" type="xs:token"/>
<xs:attribute name="trunk_group_id_b" type="xs:token"/>
<xs:attribute name="call_record_a" type="xs:string"/>
<xs:attribute name="call_record_b" type="xs:string"/>
<xs:attribute name="success_faxes" type="xs:string">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      The list of successfully received faxes, associated with current call. Urls
splitted by space.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="failed_faxes" type="xs:string">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      The list of successfully received faxes, associated with current call. Urls
splitted by space.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="url_list" type="xs:string">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Obsolete attribute.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="release_cause" type="xs:string"/>
<xs:attribute name="release_cause_isup" type="xs:string"/>
<xs:attribute name="mode" type="xs:string" use="optional"/>
</xs:complexType>
<xs:complexType name="domainOutType">
  <xs:sequence>
    <xs:element name="call" type="callType" minOccurs="0" maxOccurs="unbounded"/>

```

```

    </xs:sequence>
    <xs:attribute name="name" type="nonEmptyTokenType" use="required"/>
    <xs:attribute name="tag" type="xs:base64Binary" use="optional"/>
</xs:complexType>
<!-- End of types -->
<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="domain" type="domainType" minOccurs="1"
maxOccurs="1"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- /in -->
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="response">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain" type="domainOutType" minOccurs="1"
maxOccurs="1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
<!-- /out -->
</xs:schema

```

Пример:

Запрос: http://192.168.1.21:9999/commands/calls_history


```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_calls_history.xsd">
  <request>
    <domain name="biysk.local" from_date="13.02.2020 12:00:00" limit="10"/>
  </request>
</in>
```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_calls_history.xsd">
  <response>
    <domain name="biysk.local"
tag=
"g2gEZAAddGFnZAAHaG1zdG9yewGcZAAIZGF0ZXRpWVoAmgDYgAAB+RhAmENaANhdGEzYRBSAAAAAmGZAAFc3RhcNRRABMx
My4wMi4yMDIwIDEyOjAwOjAwaAJkAApjYwXsc190eXB1ZAADYwXsag==">
      <call is_ss_present="false" release_initiator="A" release_cause_isup="16"
        release_cause="normal" trunk_group_id_b="ems1" duration_time="0" mode="normal" addr_b="245988"
        addr_a="4957979088" trunk_group_id_a="smg-4" category_b="unknownAtThisTime"
        category_a="ordinarySubscriber" codec_a="PCMA,PCMU,G729,telephone-event" iface_b="ems1"
        iface_a="smg-4" oaddr_b="245988" oaddr_a="4957979088" stage="released" release_time="13.02.2020
        13:28:38" start_time="13.02.2020 13:28:28" parent_branch_id="" branch_id="0"
        call_id="062d8c74c686056e" call_ref="3540419372"/>
      <call is_ss_present="false" release_initiator="A" release_cause_isup="102"
        release_cause="normal" trunk_group_id_b="ems1" duration_time="0" mode="normal" addr_b="245988"
        addr_a="9629898491" trunk_group_id_a="smg-4" category_b="unknownAtThisTime"
        category_a="ordinarySubscriber" codec_a="PCMA,PCMU,G729,telephone-event" iface_b="ems1"
        iface_a="smg-4" oaddr_b="245988" oaddr_a="9629898491" stage="released" release_time="13.02.2020
        13:28:27" start_time="13.02.2020 13:28:02" parent_branch_id="" branch_id="0"
        call_id="062d8c732ec3cba0" call_ref="3431907367"/>
      <call is_ss_present="false" release_initiator="A" release_cause_isup="18"
        release_cause="aPtyDisc" trunk_group_id_b="biysk.local.ssw.gr.trunk.autoname" duration_time="0"
        mode="normal" addr_b="246077" addr_a="9606732801" trunk_group_id_a="smg-4"
        category_b="unknownAtThisTime" category_a="ordinarySubscriber" codec_a="PCMA,PCMU,G729,telephone-
        event" iface_b="ems2" iface_a="smg-4" oaddr_b="246077" oaddr_a="9606732801" stage="released"
        release_time="13.02.2020 13:20:20" start_time="13.02.2020 13:19:55" parent_branch_id=""
        branch_id="0" call_id="062d8c54b4ca187c" call_ref="1388627508"/>
      <call is_ss_present="false" release_initiator="A" release_cause_isup="16"
        release_cause="normal" trunk_group_id_b="ems1" duration_time="0" mode="normal" addr_b="245906"
        addr_a="9609374213" trunk_group_id_a="smg-4" category_b="unknownAtThisTime"
        category_a="ordinarySubscriber" codec_a="PCMA,PCMU,G729,telephone-event" iface_b="ems1"
        iface_a="smg-4" oaddr_b="245906" oaddr_a="9609374213" stage="released" release_time="13.02.2020
        13:16:00" start_time="13.02.2020 13:15:45" parent_branch_id="" branch_id="0"
        call_id="062d8c45155f92bd" call_ref="340089808"/>
      <call is_ss_present="false" release_initiator="A" release_cause_isup="102"
        release_cause="normal" trunk_group_id_b="biysk.local.ssw.gr.trunk.autoname" duration_time="0"
        mode="normal" addr_b="246653" addr_a="9069020153" trunk_group_id_a="smg-4"
        category_b="unknownAtThisTime" category_a="ordinarySubscriber" codec_a="PCMA,PCMU,G729,telephone-
        event" iface_b="ems2" iface_a="smg-4" oaddr_b="246653" oaddr_a="9069020153" stage="released"
        release_time="13.02.2020 13:07:19" start_time="13.02.2020 13:06:54" parent_branch_id=""
        branch_id="0" call_id="062d8c23ed8bcbe5" call_ref="2408417289"/>

```

```

    <call is_ss_present="false" release_initiator="A" release_cause_isup="102"
      release_cause="normal" trunk_group_id_b="biysk.local.ssw.gr.trunk.autoname" duration_time="0"
      mode="normal" addr_b="246653" addr_a="9069020031" trunk_group_id_a="smg-4"
      category_b="unknownAtThisTime" category_a="ordinarySubscriber" codec_a="PCMA,PCMU,G729,telephone-
      event" iface_b="ems2" iface_a="smg-4" oaddr_b="246653" oaddr_a="9069020031" stage="released"
      release_time="13.02.2020 13:05:16" start_time="13.02.2020 13:04:51" parent_branch_id=""
      branch_id="0" call_id="062d8c1c3c76601d" call_ref="1892446865"/>
    <call is_ss_present="false" release_initiator="A" release_cause_isup="16"
      release_cause="normal" trunk_group_id_b="ems1" duration_time="0" mode="normal" addr_b="245658"
      addr_a="4952580974" trunk_group_id_a="smg-4" category_b="unknownAtThisTime"
      category_a="localSubscriber" codec_a="PCMA,PCMU,G729,telephone-event" iface_b="ems1"
      iface_a="smg-4" oaddr_b="245658" oaddr_a="4952580974" stage="released" release_time="13.02.2020
      12:57:36" start_time="13.02.2020 12:57:25" parent_branch_id="" branch_id="0"
      call_id="062d8c005bf38e9f" call_ref="21753760"/>
    <call is_ss_present="false" release_initiator="A" release_cause_isup="38"
      release_cause="ss7Failure" trunk_group_id_b="ems1" duration_time="0" mode="normal"
      addr_b="245658" addr_a="4952580974" trunk_group_id_a="smg-4" category_b="unknownAtThisTime"
      category_a="localSubscriber" codec_a="PCMA,PCMU,G729,telephone-event" iface_b="ems1"
      iface_a="smg-4" oaddr_b="245658" oaddr_a="4952580974" stage="released" release_time="13.02.2020
      12:57:25" start_time="13.02.2020 12:57:05" parent_branch_id="" branch_id="0"
      call_id="062d8bfff16748e40" call_ref="4232474822"/>
    <call is_ss_present="false" release_initiator="A" call_record_a="http://
      system.restfs.ecss:9990/domain/biysk.local/records/
      2020_02_13/2020-02-13_12-51-56_o_9585007440-240644.wav" release_cause_isup="16"
      release_cause="normal" duration_time="2" mode="normal" addr_b="240644" addr_a="9585007440"
      trunk_group_id_a="smg-4" category_b="ordinarySubscriber" category_a="ordinarySubscriber"
      codec_b="PCMA,PCMU,G722,telephone-event,H264,H263-1998" codec_a="PCMA,telephone-event"
      iface_b="240644@biysk.local" iface_a="smg-4" oaddr_b="240644" oaddr_a="9585007440"
      stage="released" release_time="13.02.2020 12:52:01" answer_time="13.02.2020 12:51:58"
      alerting_time="13.02.2020 12:51:56" start_time="13.02.2020 12:51:56" parent_branch_id=""
      branch_id="0" call_id="062d8bebc132453" call_ref="2936672348"/>
    <call is_ss_present="false" release_initiator="A" release_cause_isup="31"
      release_cause="normal" trunk_group_id_b="ems1" duration_time="0" mode="normal" addr_b="245708"
      addr_a="9671345216" trunk_group_id_a="smg-4" category_b="unknownAtThisTime"
      category_a="ordinarySubscriber" codec_a="PCMA,PCMU,G729,telephone-event" iface_b="ems1"
      iface_a="smg-4" oaddr_b="245708" oaddr_a="9671345216" stage="released" release_time="13.02.2020
      12:51:40" start_time="13.02.2020 12:51:16" parent_branch_id="" branch_id="0"
      call_id="062d8be946df45d3" call_ref="2768690064"/>
  </domain>
</response>
</out>

```

В данном примере возвращается поле tag, которое необходимо передавать в последующем запросе, чтобы продолжить выборку. Если необходимо начать выборку с начала, необходимо в запросе опускать поле tag.

Нс_purge_calls_db

Команда удаляет устаревшие записи в БД вызовов.

Аналог команды в CoCon:

`/domain/<DOMAIN>/calls/purge`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://server:port/commands/purge_calls_db`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="hc_purge_calls_db">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Purge old records from calls_db database.
      Copyright (C) 2014, Eltex. All right reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- Types -->
  <xs:simpleType name="argType">
    <xs:restriction base="xs:string">
      <!-- DD.MM.YYYY HH:MM:SS | N day|week|month|year, N - integer -->
      <xs:pattern value="(\d{2}\.\d{2}\.\d{4}\s+\d{2}:\d{2}:\d{2}){1}|[1-9]{1}\d*\s+(day|
week|month|year){1}"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="domainType">
    <xs:attribute name="name" type="nonEmptyTokenType" use="required"/>
    <!-- <xs:attribute name="arg" type="argType" use="required"/> -->
    <xs:attribute name="arg" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="domainOutType">
    <xs:sequence>
      <xs:element name="ok" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:attribute name="affected"
            type="xs:nonNegativeInteger"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="name" type="nonEmptyTokenType" use="required"/>
  </xs:complexType>

  <!-- End of types -->

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain"

```

```

                                type="domainType"
                                minOccurs="1" maxOccurs="1"/>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                </xs:all>
            </xs:complexType>
        </xs:element>
    <!-- /in -->

    <!-- out -->
    <xs:element name="out">
        <xs:complexType>
            <xs:choice>
                <xs:element name="response">
                    <xs:complexType>
                        <xs:all>
                            <xs:element name="domain"
                                type="domainOutType"
                                minOccurs="1" maxOccurs="1"/>
                        </xs:all>
                    </xs:complexType>
                </xs:element>
                <xs:element name="error" type="errorType"
                    minOccurs="1" maxOccurs="1"/>
            </xs:choice>
        </xs:complexType>
    </xs:element>
    <!-- /out -->

</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/purge_calls_db

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="hc_purge_calls_db.xsd">
    <request>
        <domain name="biysk.local" arg="15.11.2019 15:00:00"/>
    </request>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_purge_calls_db.xsd">
  <response>
    <domain name="biysk.local">
      <ok affected="508"/>
    </domain>
  </response>
</out>
```

Hc_cc_csta_server_options - Управление настройками подключения по протоколу CSTA

- [Hc_cc_csta_server_options_clean](#)
- [Hc_cc_csta_server_options_info](#)
- [Hc_cc_csta_server_options_set](#)

[Hc_cc_csta_server_options_clean](#)

Команда сброса настроек подключения по протоколу CSTA.

Аналог команды в CoCon:

api/csta/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_csta_server_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemCleanOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_csta_server_options_clean


```

<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ecss_control_channel_options_clean.xsd">
  <request>
    <system_options>
      <system>
        <option name="use_tcp"/>
        <option name="use_ssl"/>
        <option name="tcp_port"/>
        <option name="tcp_ip"/>
        <option name="ssl_port"/>
        <option name="ssl_ip"/>
        <option name="enabled"/>
        <option name="auth"/>
      </system>
    </system_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="cc_csta_server_options_clean.xsd">
  <result>
    <system_options>
      <system>
        <success option="auth"/>
        <success option="enabled"/>
        <success option="ssl_ip"/>
        <success option="ssl_port"/>
        <success option="tcp_ip"/>
        <success option="tcp_port"/>
        <success option="use_ssl"/>
        <success option="use_tcp"/>
      </system>
    </system_options>
  </result>
</out>

```

[Hc_cc_csta_server_options_info](#)

Команда просмотра настроек подключения по протоколу CSTA.

Аналог команды в CoCon:

api/csta/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_csta_server_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- cstaServerOptions -->
  <xs:complexType name="cstaServerOptions">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="enabled" type="booleanType"/>
        <xs:attribute name="use_tcp" type="booleanType"/>
        <xs:attribute name="use_ssl" type="booleanType"/>
        <xs:attribute name="tcp_ip" type="ipAddressType"/>
        <xs:attribute name="ssl_ip" type="ipAddressType"/>
        <xs:attribute name="tcp_port" type="inetPortType"/>
        <xs:attribute name="ssl_port" type="inetPortType"/>
        <xs:attribute name="auth" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              usage:
                Login:Password;Login2:Password2
                In case of strong security restrictions password may be shown as
*****
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemInfoInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>

```

```

        <xs:element name="result">
            <xs:complexType>
                <xs:all>
                    <xs:element name="system_options" type="systemInfoOutType"/>
                </xs:all>
            </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_csta_server_options_info

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="cc_csta_server_options_info.xsd">
    <request>
        <system_options/>
    </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="cc_csta_server_options_info.xsd">
    <result>
        <system_options>
            <system>
                <options xs:type="cstaServerOptions" auth="ct.office:*****" enabled="true"
                ssl_ip="0.0.0.0" ssl_port="4722" tcp_ip="0.0.0.0" tcp_port="4721" use_ssl="true" use_tcp="true"/
            >
            </system>
        </system_options>
    </result>
</out>

```

[Hc_cc_csta_server_options_set](#)

Команда изменения настроек подключения по протоколу CSTA

Аналог команды в CoCon:

`api/csta/set`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_csta_server_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- cstaServerOptions -->
  <xs:complexType name="cstaServerOptions">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="enabled" type="booleanType"/>
        <xs:attribute name="use_tcp" type="booleanType"/>
        <xs:attribute name="use_ssl" type="booleanType"/>
        <xs:attribute name="tcp_ip" type="ipAddressType"/>
        <xs:attribute name="ssl_ip" type="ipAddressType"/>
        <xs:attribute name="tcp_port" type="inetPortType"/>
        <xs:attribute name="ssl_port" type="inetPortType"/>
        <xs:attribute name="auth" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              usage:
              Login:Password;Login2:Password2
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>

```

```

        <xs:all>
            <xs:element name="system_options" type="systemSetOutType"/>
        </xs:all>
    </xs:complexType>
</xs:element>
<xs:element name="error" type="errorType" />
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_csta_server_options_set

```

<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="cc_csta_server_options_set.xsd">
  <request>
    <system_options>
      <system>
        <options auth="test.ats:123456;domain.second:1234567" enabled="false" ssl_ip="0.0.0.0"
ssl_port="4722" tcp_ip="0.0.0.0" tcp_port="4721" use_ssl="true" use_tcp="true"
xs:type="cstaServerOptions"/>
      </system>
    </system_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="cc_csta_server_options_set.xsd">
  <result>
    <system_options>
      <system>
        <success option="use_tcp"/>
        <success option="use_ssl"/>
        <success option="tcp_port"/>
        <success option="tcp_ip"/>
        <success option="ssl_port"/>
        <success option="ssl_ip"/>
        <success option="enabled"/>
        <success option="auth"/>
      </system>
    </system_options>
  </result>
</out>

```

Нс_cc_queue_acw_status_set - Команды управления наборами статусов постобработки

- [Нс_cc_queue_acw_status_set_declare](#)
- [Нс_cc_queue_acw_status_set_remove](#)
- [Нс_cc_queue_acw_status_set_info](#)
- [Нс_cc_queue_acw_status_add_status](#)
- [Нс_cc_queue_acw_status_set_remove_status](#)

Нс_cc_queue_acw_status_set_declare

Команда добавления нового набора статусов постобработки.

Аналог команды в CoSop:

/domain/<DOMAIN>/cc/queue/acw/status-set/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_acw_status_set_declare

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <!--statusType-->
  <xs:complexType name="statusSetType">
    <xs:attribute name="id" use="required"/>
  </xs:complexType>
  <!--requestType-->
  <xs:complexType name="requestType">
    <!--request body-->
    <xs:sequence>
      <xs:element name="set" type="statusSetType"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!--responseType-->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="ok"/>
    </xs:sequence>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="responseType"/>
        <!--bad. command failed-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_acw_status_set_declare

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_cc_queue_acw_status_set_declare.xsd">
  <request domain="test.domain">
    <set id="test"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="hc_cc_queue_acw_status_set_declare.xsd">
  <response>
    <ok/>
  </response>
</out>
```

hc_cc_queue_acw_status_set_remove

Команда добавления/удаления статуса постобработки.

Аналог команды в CoCon:

/domain/<DOMAIN>/cc/queue/acw/status-set/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_acw_status_set_remove

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <!--statusType-->
  <xs:complexType name="statusType">
    <xs:attribute name="id" use="required"/>
  </xs:complexType>
  <!--requestType-->
  <xs:complexType name="requestType">
    <!--request body-->
    <xs:sequence>
      <xs:element name="status" type="statusType"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!--responseType-->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="ok"/>
    </xs:sequence>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="responseType"/>
        <!--bad. command failed-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_acw_status_set_remove

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_cc_queue_acw_status_set_remove.xsd">
  <request domain="test.domain">
    <status id="test"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cc_queue_acw_status_remove.xsd">
  <response>
    <ok/>
  </response>
</out>
```

[Hc_cc_queue_acw_status_set_info](#)

Команда просмотра списка статусов постобработки.

Аналог команды в CoCon:

/domain/<DOMAIN>/cc/queue/acw/status-set/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_acw_status_set_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="statusType">
    <xs:attribute name="id" use="required"/>
  </xs:complexType>

  <!-- statusType -->
  <xs:complexType name="statusSetType">
    <xs:sequence>
      <xs:element name="status" type="statusType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" use="required"/>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="set" type="statusSetType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="responseType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_acw_status_set_info

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_cc_queue_acw_status_set_info.xsd">
  <request domain="test.domain"/>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="hc_cc_queue_acw_status_set_info.xsd">
  <response>
    <set id="test">
      <status id="test"/>
    </set>
  </response>
</out>
```

hc_cc_queue_acw_status_add_status

Команда добавления нового статуса постобработки в набор.

Аналог команды в CoCon:

/domain/<DOMAIN>/cc/queue/acw/status-set/add-status

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_acw_status_set_add_status

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <!--statusType-->
  <xs:complexType name="statusType">
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>
  <!--statusSetType-->
  <xs:complexType name="statusSetType">
    <xs:sequence>
      <xs:element name="status" type="statusType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>
  <!--requestType-->
  <xs:complexType name="requestType">
    <!--request body-->
    <xs:sequence>
      <xs:element name="set" type="statusSetType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!--warningType-->
  <xs:complexType name="warningType">
    <xs:attribute name="set" type="xs:string" use="required"/>
    <xs:attribute name="status" type="xs:string" use="required"/>
    <xs:attribute name="body" type="errorReasonType"/>
  </xs:complexType>
  <!--warningsType-->
  <xs:complexType name="warningsType">
    <xs:sequence>
      <xs:element name="warning" type="warningType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!--responseType-->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok"/>
      <xs:element name="warnings" type="warningsType"/>
    </xs:choice>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence maxOccurs="unbounded">
        <xs:choice>
          <xs:element name="response" type="responseType"/>
          <!--bad. command failed!-->
          <xs:element name="error" type="errorType"/>
        </xs:choice>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_acw_status_set_add_status

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_cc_queue_acw_status_set_add_status.xsd">
  <request domain="test.domain">
    <set id="test">
      <status id="test"/>
    </set>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="hc_cc_queue_acw_status_set_add_status.xsd">
  <response>
    <ok/>
  </response>
</out>

```

[Hc_cc_queue_acw_status_set_remove_status](#)

Команда удаляет статус постобработки из набора.

Аналог команды в CoCon:

/domain/<DOMAIN>/cc/queue/acw/status-set/remove-status

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_acw_status_set_remove_status

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <!--statusType-->
  <xs:complexType name="statusType">
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>
  <!--statusSetType-->
  <xs:complexType name="statusSetType">
    <xs:sequence>
      <xs:element name="status" type="statusType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>
  <!--requestType-->
  <xs:complexType name="requestType">
    <!--request body-->
    <xs:sequence>
      <xs:element name="set" type="statusSetType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!--warningType-->
  <xs:complexType name="warningType">
    <xs:attribute name="set" type="xs:string" use="required"/>
    <xs:attribute name="status" type="xs:string" use="required"/>
    <xs:attribute name="body" type="errorReasonType"/>
  </xs:complexType>
  <!--warningsType-->
  <xs:complexType name="warningsType">
    <xs:sequence>
      <xs:element name="warning" type="warningType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!--responseType-->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok"/>
      <xs:element name="warnings" type="warningsType"/>
    </xs:choice>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="responseType"/>
        <!--bad. command failed!-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_acw_status_set_remove_status

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_cc_queue_acw_status_set_remove_status.xsd">
  <request domain="test.domain">
    <set id="test">
      <status id="test"/>
    </set>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="hc_cc_queue_acw_status_set_remove_status.xsd">
  <response>
    <ok/>
  </response>
</out>

```

Нс_cc_queue_acw_status - Команды управления статусами постобработки

- [Нс_cc_queue_acw_status_declare](#)
- [Нс_cc_queue_acw_status_remove](#)
- [Нс_cc_queue_acw_status_info](#)

- [Hc_cc_queue_acw_status_add_label](#)
- [Hc_cc_queue_acw_status_remove_label](#)

[Hc_cc_queue_acw_status_declare](#)

Команда добавления нового статуса постобработки.

Аналог команды в CoCon:

/domain/<DOMAIN>/cc/queue/acw/status/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_acw_status_declare

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <!--statusType-->
  <xs:complexType name="statusType">
    <xs:attribute name="id" use="required"/>
  </xs:complexType>
  <!--requestType-->
  <xs:complexType name="requestType">
    <!--request body-->
    <xs:sequence>
      <xs:element name="status" type="statusType"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!--responseType-->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="ok"/>
    </xs:sequence>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="responseType"/>
        <!--bad. command failed-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_acw_status_declare

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_cc_queue_acw_status_declare.xsd">
  <request domain="test.domain">
    <status id="test"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cc_queue_acw_status_declare.xsd">
  <response>
    <ok/>
  </response>
</out>
```

[Hc_cc_queue_acw_status_remove](#)

Команда добавления/удаления статуса постобработки.

Аналог команды в CoCon:

/domain/<DOMAIN>/cc/queue/acw/status/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_acw_status_remove

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <!--statusType-->
  <xs:complexType name="statusType">
    <xs:attribute name="id" use="required"/>
  </xs:complexType>
  <!--requestType-->
  <xs:complexType name="requestType">
    <!--request body-->
    <xs:sequence>
      <xs:element name="status" type="statusType"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!--responseType-->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="ok"/>
    </xs:sequence>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="responseType"/>
        <!--bad. command failed:-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_acw_status_remove

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_cc_queue_acw_status_remove.xsd">
  <request domain="test.domain">
    <status id="test"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_cc_queue_acw_status_remove.xsd">
  <request domain="test.domain">
    <status id="test"/>
  </request>
</in>
```

[Hc_cc_queue_acw_status_info](#)

Команда просмотра списка статусов постобработки.

Аналог команды в CoCon:

/domain/<DOMAIN>/cc/queue/acw/status/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_acw_status_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="translateType">
    <xs:attribute name="lang" use="required"/>
    <xs:attribute name="value" use="required"/>
  </xs:complexType>

  <!--statusType-->
  <xs:complexType name="statusType">
    <xs:sequence>
      <xs:element name="translate" type="translateType" minOccurs="0" maxOccurs="unbounded"
/>
    </xs:sequence>
    <xs:attribute name="id" use="required"/>
  </xs:complexType>
  <!--responseType-->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="status" type="statusType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="responseType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/cc_queue_acw_status_info

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_cc_queue_acw_status_info.xsd">
  <request domain="test.domain"/>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
  <out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="hc_cc_queue_acw_status_info.xsd">
    <response>
      <status id="test">
        <translate lang="en" value="test"/>
      </status>
    </response>
  </out>
```

hc_cc_queue_acw_status_add_label

Команда добавления нового статуса постобработки.

Аналог команды в CoCon:

/domain/<DOMAIN>/cc/queue/acw/status/add-label

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_acw_status_add_label

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <!--statusType-->
  <xs:complexType name="statusType">
    <xs:attribute name="id" use="required"/>
    <xs:attribute name="lang" use="required"/>
    <xs:attribute name="value" use="required"/>
  </xs:complexType>
  <!--requestType-->
  <xs:complexType name="requestType">
    <!--request body-->
    <xs:sequence>
      <xs:element name="status" type="statusType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!--warningType-->
  <xs:complexType name="warningType">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="lang" type="xs:string" use="required"/>
    <xs:attribute name="body" type="errorReasonType"/>
  </xs:complexType>
  <!--warningsType-->
  <xs:complexType name="warningsType">
    <xs:sequence>
      <xs:element name="warning" type="warningType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!--responseType-->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok"/>
      <xs:element name="warnings" type="warningsType"/>
    </xs:choice>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>

```

```

    <xs:choice>
      <xs:element name="response" type="responseType"/>
      <!--bad. command failed!-->
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_acw_status_add_label

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_cc_queue_acw_status_add_label.xsd">
  <request domain="test.domain">
    <status id="test" lang="ru" value="тест"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="hc_cc_queue_acw_status_add_label.xsd">
  <response>
    <ok/>
  </response>
</out>

```

hc_cc_queue_acw_status_remove_label

Команда добавления нового статуса постобработки.

Аналог команды в CoSop:

/domain/<DOMAIN>/cc/queue/acw/status/remove-label

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_acw_status_remove_label

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <!--statusType-->
  <xs:complexType name="statusType">
    <xs:attribute name="id" use="required"/>
    <xs:attribute name="lang" use="required"/>
  </xs:complexType>
  <!--requestType-->
  <xs:complexType name="requestType">
    <!--request body-->
    <xs:sequence>
      <xs:element name="status" type="statusType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!--warningType-->
  <xs:complexType name="warningType">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="lang" type="xs:string" use="required"/>
    <xs:attribute name="body" type="errorReasonType"/>
  </xs:complexType>
  <!--warningsType-->
  <xs:complexType name="warningsType">
    <xs:sequence>
      <xs:element name="warning" type="warningType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!--responseType-->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok"/>
      <xs:element name="warnings" type="warningsType"/>
    </xs:choice>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>

```

```

        <xs:element name="response" type="responseType"/>
        <!--bad. command failed!-->
        <xs:element name="error" type="errorType"/>
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_acw_status_remove_label

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_cc_queue_acw_status_remove_label.xsd">
    <request domain="test.domain">
        <status id="test" lang="ru"/>
    </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="hc_cc_queue_acw_status_remove_label.xsd">
    <response>
        <ok/>
    </response>
</out>

```

Нс_cc_queue_agent_auxwork_reason - Команды управления причинами технического перерыва агента Call-центра

- [Нс_cc_queue_agent_auxwork_reason_add](#)
- [Нс_cc_queue_agent_auxwork_reason_clean](#)
- [Нс_cc_queue_agent_auxwork_reason_info](#)
- [Нс_cc_queue_agent_auxwork_reason_remove](#)
- [Нс_cc_queue_agent_auxwork_reason_set](#)

[Нс_cc_queue_agent_auxwork_reason_add](#)

Команда добавления новой причины технического перерыва агентов.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/agent/auxwork/reason/add

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_agent_auxwork_reason_add

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<xs:include schemaLocation="error.xsd"/>
<!--reasonType-->
<xs:complexType name="reasonType">
  <xs:attribute name="id" use="required"/>
</xs:complexType>
<!--requestType-->
<xs:complexType name="requestType">
  <!--request body-->
  <xs:sequence>
    <xs:element name="reason" type="reasonType"/>
  </xs:sequence>
  <xs:attribute name="domain" type="xs:string" use="required"/>
</xs:complexType>
<!--responseType-->
<xs:complexType name="responseType">
  <xs:sequence>
    <xs:element name="ok"/>
  </xs:sequence>
</xs:complexType>
<!--in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request" type="requestType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!--out-->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="response" type="responseType"/>
      <!--bad. command failed-->
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_agent_auxwork_reason_add

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_cc_queue_agent_auxwork_reason_add.xsd">
  <request domain="biysk.local">
    <reason id="empty_reason"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cc_queue_agent_auxwork_reason_add.xsd">
  <response>
    <ok/>
  </response>
</out>
```

[Hc_cc_queue_agent_auxwork_reason_clean](#)

Команда удаления перевода причины перерыва агентов.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/agent/auxwork/reason/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_agent_auxwork_reason_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<xs:include schemaLocation="error.xsd"/>
<!--reasonType-->
<xs:complexType name="reasonType">
  <xs:attribute name="id" use="required"/>
  <xs:attribute name="lang" use="required"/>
</xs:complexType>
<!--requestType-->
<xs:complexType name="requestType">
  <!--request body-->
  <xs:sequence>
    <xs:element name="reason" type="reasonType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="domain" type="xs:string" use="required"/>
</xs:complexType>
<!--warningType-->
<xs:complexType name="warningType">
  <xs:attribute name="id" type="xs:string" use="required"/>
  <xs:attribute name="lang" type="xs:string" use="required"/>
  <xs:attribute name="body" type="errorReasonType"/>
</xs:complexType>
<!--warningsType-->
<xs:complexType name="warningsType">
  <xs:sequence>
    <xs:element name="warning" type="warningType" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!--responseType-->
<xs:complexType name="responseType">
  <xs:choice>
    <xs:element name="ok"/>
    <xs:element name="warnings" type="warningsType"/>
  </xs:choice>
</xs:complexType>
<!--in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request" type="requestType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!--out-->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="response" type="responseType"/>
      <!--bad. command failed-->
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

```

```
</xs:complexType>  
</xs:element>
```

Нс_cc_queue_agent_auxwork_reason_info

Команда получения информации обо всех причинах перерыва агентов.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/agent/auxwork/reason/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_agent_auxwork_reason_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="translateType">
    <xs:attribute name="lang" use="required"/>
    <xs:attribute name="value" use="required"/>
  </xs:complexType>

  <!-- reasonType -->
  <xs:complexType name="reasonType">
    <xs:sequence>
      <xs:element name="translate" type="translateType" minOccurs="0" maxOccurs="unbounded"
/>
    </xs:sequence>
    <xs:attribute name="id" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="reason" type="reasonType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="responseType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/cc_queue_agent_auxwork_reason_info

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_cc_queue_agent_auxwork_reason_info.xsd">
  <request domain="biysk.local"/>
</in>
```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cc_queue_agent_auxwork_reason_info.xsd">
  <response>
    <reason id="124">
      <translate lang="en" value="124"/>
      <translate lang="ru" value="124"/>
    </reason>
    <reason id="locality_transfer">
      <translate lang="en" value="Transferring to another locality"/>
      <translate lang="ru" value="Передача в другой населенный пункт"/>
    </reason>
    <reason id="dinner">
      <translate lang="en" value="Dinner"/>
      <translate lang="ru" value="Обед"/>
    </reason>
    <reason id="report">
      <translate lang="en" value="Report"/>
      <translate lang="ru" value="Отчет"/>
    </reason>
    <reason id="remote">
      <translate lang="en" value="Remote admin"/>
      <translate lang="ru" value="Удаленный доступ"/>
    </reason>
    <reason id="handle_call">
      <translate lang="en" value="Call handling"/>
      <translate lang="ru" value="Обработка вызова"/>
    </reason>
    <reason id="rest">
      <translate lang="en" value="Rest"/>
      <translate lang="ru" value="Отдых"/>
    </reason>
    <reason id="escalation">
      <translate lang="en" value="Transfer to a specialist"/>
      <translate lang="ru" value="Перевод специалисту"/>
    </reason>
    <reason id="washroom">
      <translate lang="ru" value="туалет"/>
    </reason>
    <reason id="empty_reason"/>
    <reason id="visit"/>
    <reason id="busy">
      <translate lang="en" value="Busy"/>
      <translate lang="ru" value="Занят"/>
    </reason>
    <reason id="consult">
      <translate lang="en" value="Consult"/>
      <translate lang="ru" value="Консультация"/>
    </reason>
  </response>
</out>

```

```
</reason>
<reason id="callback">
  <translate lang="en" value="Callback"/>
  <translate lang="ru" value="Обратный звонок"/>
</reason>
</response>
</out>
```

[Hc_cc_queue_agent_auxwork_reason_remove](#)

Команда удаления причины технического перерыва агентов.

Аналог команды в CoСon:

domain/<DOMAIN>/cc/agent/auxwork/reason/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_agent_auxwork_reason_remove

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <!-- reasonType -->
  <xs:complexType name="reasonType">
    <xs:attribute name="id" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <!-- request body -->
    <xs:sequence>
      <xs:element name="reason" type="reasonType"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="ok"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="responseType"/>
        <!-- bad. command failed -->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_agent_auxwork_reason_remove

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_cc_queue_agent_auxwork_reason_remove.xsd">
  <request domain="biysk.local">
    <reason id="visit"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cc_queue_agent_auxwork_reason_remove.xsd">
  <response>
    <ok/>
  </response>
</out>
```

[Hc_cc_queue_agent_auxwork_reason_set](#)

Команда изменения причины технического перерыва агентов.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/agent/auxwork/reason/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_agent_auxwork_reason_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <!-- reasonType -->
  <xs:complexType name="reasonType">
    <xs:attribute name="id" use="required"/>
    <xs:attribute name="lang" use="required"/>
    <xs:attribute name="value" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <!-- request body -->
    <xs:sequence>
      <xs:element name="reason" type="reasonType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- warningType -->
  <xs:complexType name="warningType">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="lang" type="xs:string" use="required"/>
    <xs:attribute name="body" type="errorReasonType"/>
  </xs:complexType>
  <!-- warningsType -->
  <xs:complexType name="warningsType">
    <xs:sequence>
      <xs:element name="warning" type="warningType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok"/>
      <xs:element name="warnings" type="warningsType"/>
    </xs:choice>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>

```

```

    <xs:choice>
      <xs:element name="response" type="responseType"/>
      <!--bad. command failed!-->
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_agent_auxwork_reason_set

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_cc_queue_agent_auxwork_reason_set.xsd">
  <request domain="biysk.local">
    <reason id="empty_reason" lang="ru" value="Устал"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_cc_queue_agent_auxwork_reason_set.xsd">
  <response>
    <ok/>
  </response>
</out>

```

Нс_cc_queue_agent_profile - Команды изменения причин технического перерыва для профиля агента call-центра

- [Нс_cc_queue_agent_profile_add](#)
- [Нс_cc_queue_agent_profile_clean](#)
- [Нс_cc_queue_agent_profile_info](#)
- [Нс_cc_queue_agent_profile_remove](#)
- [Нс_cc_queue_agent_profile_set](#)

Ис_cc_queue_agent_profile_add

Команда добавления нового профиля причин технического перерыва агента.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/agent/auxwork/profile/add

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_agent_profile_add

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<xs:include schemaLocation="error.xsd"/>
<!--statusType-->
<xs:complexType name="profileType">
  <xs:attribute name="id" use="required"/>
</xs:complexType>
<!--requestType-->
<xs:complexType name="requestType">
  <!--request body-->
  <xs:sequence>
    <xs:element name="profile" type="profileType"/>
  </xs:sequence>
  <xs:attribute name="domain" type="xs:string" use="required"/>
</xs:complexType>
<!--responseType-->
<xs:complexType name="responseType">
  <xs:sequence>
    <xs:element name="ok"/>
  </xs:sequence>
</xs:complexType>
<!--in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request" type="requestType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!--out-->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="response" type="responseType"/>
      <!--bad. command failed-->
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_agent_profile_add

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_cc_queue_agent_profile_add.xsd">
  <request domain="biysk.local">
    <profile id="test_profile"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cc_queue_agent_profile_add.xsd">
  <response>
    <ok/>
  </response>
</out>
```

[Hc_cc_queue_agent_profile_clean](#)

Команда удаления причины перерыва из профиля агента.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/agent/auxwork/profile/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_agent_profile_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<xs:include schemaLocation="error.xsd"/>
<!--reasonType-->
<xs:complexType name="reasonType">
  <xs:attribute name="id" type="xs:string" use="required"/>
</xs:complexType>
<!--profileType-->
<xs:complexType name="profileType">
  <xs:sequence>
    <xs:element name="reason" type="reasonType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required"/>
</xs:complexType>
<!--requestType-->
<xs:complexType name="requestType">
  <!--request body-->
  <xs:sequence>
    <xs:element name="profile" type="profileType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="domain" type="xs:string" use="required"/>
</xs:complexType>
<!--warningType-->
<xs:complexType name="warningType">
  <xs:attribute name="profile" type="xs:string" use="required"/>
  <xs:attribute name="reason" type="xs:string" use="required"/>
  <xs:attribute name="body" type="errorReasonType"/>
</xs:complexType>
<!--warningsType-->
<xs:complexType name="warningsType">
  <xs:sequence>
    <xs:element name="warning" type="warningType" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!--responseType-->
<xs:complexType name="responseType">
  <xs:choice>
    <xs:element name="ok"/>
    <xs:element name="warnings" type="warningsType"/>
  </xs:choice>
</xs:complexType>
<!--in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request" type="requestType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!--out-->
<xs:element name="out">

```



```

<xs:complexType>
  <xs:choice>
    <xs:element name="response" type="responseType"/>
    <!--bad. command failed!-->
    <xs:element name="error" type="errorType"/>
  </xs:choice>
</xs:complexType>
</xs:element>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_agent_profile_clean

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_cc_queue_agent_profile_clean.xsd">
  <request domain="biysk.local">
    <profile id="test_profile">
      <reason id="test_reason"/>
      <reason id="undefined"/>
    </profile>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cc_queue_agent_profile_clean.xsd">
  <response>
    <ok/>
  </response>
</out>

```

hc_cc_queue_agent_profile_info

Команда получения информации обо всех профилях причин перерыва агентов.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/agent/auxwork/profile/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_agent_profile_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="reasonType">
    <xs:attribute name="id" use="required"/>
  </xs:complexType>

  <!--reasonType-->
  <xs:complexType name="profileType">
    <xs:sequence>
      <xs:element name="reason" type="reasonType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" use="required"/>
  </xs:complexType>
  <!--responseType-->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="profile" type="profileType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="responseType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/cc_queue_agent_profile_info

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_cc_queue_agent_profile_info.xsd">
  <request domain="biysk.local"/>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cc_queue_agent_profile_info.xsd">
  <response>
    <profile id="HelpMe">
      <reason id="busy"/>
      <reason id="callback"/>
      <reason id="consult"/>
      <reason id="dinner"/>
      <reason id="handle_call"/>
      <reason id="locality_transfer"/>
      <reason id="remote"/>
      <reason id="report"/>
      <reason id="rest"/>
    </profile>
    <profile id="103">
      <reason id="busy"/>
      <reason id="dinner"/>
      <reason id="124"/>
      <reason id="handle_call"/>
      <reason id="callback"/>
      <reason id="rest"/>
      <reason id="locality_transfer"/>
    </profile>
    <profile id="test_profile">
      <reason id="busy"/>
    </profile>
    <profile id="default">
      <reason id="124"/>
      <reason id="busy"/>
      <reason id="callback"/>
      <reason id="consult"/>
      <reason id="dinner"/>
      <reason id="empty_reason"/>
      <reason id="handle_call"/>
      <reason id="locality_transfer"/>
      <reason id="remote"/>
      <reason id="report"/>
      <reason id="rest"/>
    </profile>
    <profile id="test_reason">
      <reason id="busy"/>
    </profile>
  </response>
</out>
```

[Hc_cc_queue_agent_profile_remove](#)

Команда удаления профиля причин технического перерыва агента.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/agent/auxwork/profile/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_agent_profile_remove

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<xs:include schemaLocation="error.xsd"/>
  <!--statusType-->
  <xs:complexType name="profileType">
    <xs:attribute name="id" use="required"/>
  </xs:complexType>
  <!--requestType-->
  <xs:complexType name="requestType">
    <!--request body-->
    <xs:sequence>
      <xs:element name="profile" type="profileType"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!--responseType-->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="ok"/>
    </xs:sequence>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="responseType"/>
        <!--bad. command failed-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_agent_profile_remove

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_cc_queue_agent_profile_remove.xsd">
  <request domain="biysk.local">
    <profile id="test_profile"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cc_queue_agent_profile_remove.xsd">
  <response>
    <ok/>
  </response>
</out>
```

[Hc_cc_queue_agent_profile_set](#)

Команда изменения профиля причин технического перерыва агента.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/agent/auxwork/reason/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cc_queue_agent_profile_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<xs:include schemaLocation="error.xsd"/>
<!--reasonType-->
<xs:complexType name="reasonType">
  <xs:attribute name="id" type="xs:string" use="required"/>
</xs:complexType>
<!--profileType-->
<xs:complexType name="profileType">
  <xs:sequence>
    <xs:element name="reason" type="reasonType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required"/>
</xs:complexType>
<!--requestType-->
<xs:complexType name="requestType">
  <!--request body-->
  <xs:sequence>
    <xs:element name="profile" type="profileType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="domain" type="xs:string" use="required"/>
</xs:complexType>
<!--warningType-->
<xs:complexType name="warningType">
  <xs:attribute name="profile" type="xs:string" use="required"/>
  <xs:attribute name="reason" type="xs:string" use="required"/>
  <xs:attribute name="body" type="errorReasonType"/>
</xs:complexType>
<!--warningsType-->
<xs:complexType name="warningsType">
  <xs:sequence>
    <xs:element name="warning" type="warningType" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!--responseType-->
<xs:complexType name="responseType">
  <xs:choice>
    <xs:element name="ok"/>
    <xs:element name="warnings" type="warningsType"/>
  </xs:choice>
</xs:complexType>
<!--in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request" type="requestType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!--out-->
<xs:element name="out">

```



```

<xs:complexType>
  <xs:sequence maxOccurs="unbounded">
    <xs:choice>
      <xs:element name="response" type="responseType"/>
      <!--bad. command failed!-->
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:sequence>
</xs:complexType>
</xs:element>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cc_queue_agent_profile_set

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_cc_queue_agent_profile_set.xsd">
  <request domain="biysk.local">
    <profile id="HelpMe">
      <reason id="test_reason"/>
      <reason id="washroom"/>
    </profile>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cc_queue_agent_profile_set.xsd">
  <response>
    <ok/>
  </response>
</out>

```

Нс_cfc_options - Управление настройками сервиса CFC

- [Нс_cfc_options_clean](#)
- [Нс_cfc_options_info](#)
- [Нс_cfc_options_set](#)

Ис_cfc_options_clean

Команда сброса настроек сервиса CFC.

Аналог команды в CoCon:

`/cluster/core/<CORE>/cfc/clean`

`/domain/<DOMAIN>/cfc/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/cfc_options_clean`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean properties of CFC service.
      Usage: http://server:9999/commands/cfc_options_clean.
      Implementation: hc_cfc_options_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/cfc_options_clean

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cfc_options_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="authorisation_failure"/>
        <option name="enable"/>
        <option name="external_acp_causes"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cfc_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="authorisation_failure"/>
        <success option="enable"/>
        <success option="external_acp_causes"/>
      </domain>
    </domain_options>
  </result>
</out>

```

[Hc_cfc_options_info](#)

Команда просмотра информации о настройках сервиса CFC.

Аналог команды в CoCon:

`/cluster/core/<CORE>/cfc/info`

`/domain/<DOMAIN>/cfc/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cfc_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of CFC service.
      Usage: http://server:9999/commands/cfc_options_info.
      Implementation: hc_cfc_options_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
      Tone id format:
        tone://?f2=425&off_msec=4000&on_msec=1000&f1=0
        system://PATH/FILE.wav
        domain://PATH/FILE.wav
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="CfcOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="pty_abandon" type="xs:string" use="optional"/>
        <xs:attribute name="pty_disc" type="xs:string" use="optional"/>
        <xs:attribute name="authorisation_failure" type="xs:string" use="optional"/>
        <xs:attribute name="bpty_busy_ndub" type="xs:string" use="optional"/>
        <xs:attribute name="bpty_busy_udub" type="xs:string" use="optional"/>
        <xs:attribute name="bpty_disc" type="xs:string" use="optional"/>
        <xs:attribute name="bpty_no_answer" type="xs:string" use="optional"/>
        <xs:attribute name="call_being_forwarded" type="xs:string" use="optional"/>
        <xs:attribute name="called_party_rejected" type="xs:string" use="optional"/>
        <xs:attribute name="collect_information_failure" type="xs:string" use="optional"/>
        <xs:attribute name="conversation_timeout" type="xs:string" use="optional"/>
        <xs:attribute name="do_not_disturb" type="xs:string" use="optional"/>
        <xs:attribute name="destination_out_of_order" type="xs:string" use="optional"/>
        <xs:attribute name="anonymity_disallowed" type="xs:string" use="optional"/>
        <xs:attribute name="enable" type="xs:boolean" use="optional"/>
        <xs:attribute name="external_acp_causes" type="xs:string" use="optional"/>
        <xs:attribute name="external_isup_causes" type="xs:string" use="optional"/>
        <xs:attribute name="foreground" type="xs:boolean" use="optional"/>
        <xs:attribute name="invalid_collected_information" type="xs:string" use="optional"/>
        <xs:attribute name="invalid_number" type="xs:string" use="optional"/>
        <xs:attribute name="no_circuit_available" type="xs:string" use="optional"/>
        <xs:attribute name="no_requested_circuit_available" type="xs:string" use="optional"/>
        <xs:attribute name="normal" type="xs:string" use="optional"/>
        <xs:attribute name="not_reachable" type="xs:string" use="optional"/>
        <xs:attribute name="number_incomplete" type="xs:string" use="optional"/>
        <xs:attribute name="o_no_answer" type="xs:string" use="optional"/>
        <xs:attribute name="origination_denied" type="xs:string" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

<xs:attribute name="route_failure1" type="xs:string" use="optional"/>
<xs:attribute name="route_failure2" type="xs:string" use="optional"/>
<xs:attribute name="route_select_failure" type="xs:string" use="optional"/>
<xs:attribute name="ss7_failure" type="xs:string" use="optional"/>
<xs:attribute name="system_failure" type="xs:string" use="optional"/>
<xs:attribute name="t_exception" type="xs:string" use="optional"/>
<xs:attribute name="termination_denied" type="xs:string" use="optional"/>
<xs:attribute name="unassigned_number" type="xs:string" use="optional"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoInType" minOccurs="1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/cfc_options_info

```
<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_cfc_options_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>
```

Код ответа:

200

Ответ:


```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cfc_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="CfcOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="CfcOptionsType" unassigned_number="system://sounds/
ai_notconnected.wav" termination_denied="system://sounds/ai_tempdenied.wav"
t_exception="system://sounds/ai_system_problem.wav" system_failure="system://sounds/
ai_system_problem.wav" ss7_failure="system://sounds/ai_system_problem.wav"
route_select_failure="system://sounds/ai_out_of_order.wav" route_failure2="" route_failure1=""
origination_denied="system://sounds/ai_notaccess.wav" o_no_answer="system://sounds/
ai_no_answer.wav" number_incomplete="system://sounds/ai_invalidnumber.wav"
not_reachable="system://sounds/ai_not_reachable.wav" normal=""
no_requested_circuit_available="system://sounds/ai_badroute.wav" no_circuit_available="system://
sounds/ai_badroute.wav" invalid_number="system://sounds/ai_invalidnumber.wav"
invalid_collected_information="system://sounds/ai_wrong_number.wav" foreground="false"
external_isup_causes="1:system://sounds/ai_number_absent.wav;2:system://sounds/
ai_wrong_number.wav;3:system://sounds/ai_wrong_number.wav;18:system://sounds/
ai_not_reachable.wav;19:system://sounds/ai_no_answer.wav;20:system://sounds/ai_notconnected.wav;
21:system://sounds/ai_notaccess.wav;22:system://sounds/ai_number_not_ready.wav;27:system://
sounds/ai_out_of_order.wav;28:system://sounds/ai_invalidnumber.wav;34:system://sounds/
ai_overload.wav;44:system://sounds/ai_overload.wav;38:system://sounds/ai_error.wav;41:system://
sounds/ai_error.wav;42:system://sounds/ai_system_problem.wav;102:system://sounds/ai_timeout.wav"
external_acp_causes="" enable="true" do_not_disturb="" destination_out_of_order="system://
sounds/ai_error.wav" conversation_timeout="system://sounds/ai_conversation_timeout.wav"
collect_information_failure="system://sounds/ai_wrong_number.wav" called_party_rejected=""
call_being_forwarded="" bpty_no_answer="system://sounds/ai_no_answer.wav" bpty_disc=""
bpty_busy_udub="" bpty_busy_ndub="" authorisation_failure="system://sounds/ai_wrong_number.wav"
apty_disc="" apty_abandon="" anonymity_disallowed=""/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_cfc_options_set

Команда изменения свойств сервиса CFC.

Аналог команды в CoCon:

```
/cluster/core/<CORE>/cfc/set
```

```
/domain/<DOMAIN>/cfc/set
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cfc_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set CFC properties
      Usage: http://server:9999/commands/cfc_options_set.
      Implementation: hc_cfc_options_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
      Tone id format:
        tone://?f2=425&off_msec=4000&on_msec=1000&f1=0
        system://PATH/FILE.wav
        domain://PATH/FILE.wav
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="CfcOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="pty_abandon" type="xs:string" use="optional"/>
        <xs:attribute name="pty_disc" type="xs:string" use="optional"/>
        <xs:attribute name="authorisation_failure" type="xs:string" use="optional"/>
        <xs:attribute name="bpty_busy_ndub" type="xs:string" use="optional"/>
        <xs:attribute name="bpty_busy_udub" type="xs:string" use="optional"/>
        <xs:attribute name="bpty_disc" type="xs:string" use="optional"/>
        <xs:attribute name="bpty_no_answer" type="xs:string" use="optional"/>
        <xs:attribute name="call_being_forwarded" type="xs:string" use="optional"/>
        <xs:attribute name="called_party_rejected" type="xs:string" use="optional"/>
        <xs:attribute name="collect_information_failure" type="xs:string" use="optional"/>
        <xs:attribute name="conversation_timeout" type="xs:string" use="optional"/>
        <xs:attribute name="do_not_disturb" type="xs:string" use="optional"/>
        <xs:attribute name="destination_out_of_order" type="xs:string" use="optional"/>
        <xs:attribute name="anonymity_disallowed" type="xs:string" use="optional"/>
        <xs:attribute name="enable" type="xs:boolean" use="optional"/>
        <xs:attribute name="external_acp_causes" type="xs:string" use="optional"/>
        <xs:attribute name="external_isup_causes" type="xs:string" use="optional"/>
        <xs:attribute name="foreground" type="xs:boolean" use="optional"/>
        <xs:attribute name="invalid_collected_information" type="xs:string" use="optional"/>
        <xs:attribute name="invalid_number" type="xs:string" use="optional"/>
        <xs:attribute name="no_circuit_available" type="xs:string" use="optional"/>
        <xs:attribute name="no_requested_circuit_available" type="xs:string" use="optional"/>
        <xs:attribute name="normal" type="xs:string" use="optional"/>
        <xs:attribute name="not_reachable" type="xs:string" use="optional"/>
        <xs:attribute name="number_incomplete" type="xs:string" use="optional"/>
        <xs:attribute name="o_no_answer" type="xs:string" use="optional"/>
        <xs:attribute name="origination_denied" type="xs:string" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

<xs:attribute name="route_failure1" type="xs:string" use="optional"/>
<xs:attribute name="route_failure2" type="xs:string" use="optional"/>
<xs:attribute name="route_select_failure" type="xs:string" use="optional"/>
<xs:attribute name="ss7_failure" type="xs:string" use="optional"/>
<xs:attribute name="system_failure" type="xs:string" use="optional"/>
<xs:attribute name="t_exception" type="xs:string" use="optional"/>
<xs:attribute name="termination_denied" type="xs:string" use="optional"/>
<xs:attribute name="unassigned_number" type="xs:string" use="optional"/>
  </xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/cfc_options_set

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cfc_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="CfcOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="CfcOptionsType" enable="true" normal="" foreground="true"
        not_reachable="ai_not_reachable" external_isup_causes="1:ai_number_absent;21:ai_not_pay"
        external_acp_causes="test:ai_wrong_number" />
      </domain>
    </domain_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cfc_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="enable"/>
        <success option="external_acp_causes"/>
        <success option="external_isup_causes"/>
        <success option="foreground"/>
        <success option="normal"/>
        <success option="not_reachable"/>
      </domain>
    </domain_options>
  </result>
</out>

```

hc_change_context - Команда импорта контекста маршрутизации

Команда предназначена для импорта контекста в домен (в случае, если контекст с данным именем уже существует, он будет обновлен; если контекст с данным именем не существует, он будет создан).

Метод http запроса

POST

Шаблон http запроса

http://host:port/commands/change_context

Код http ответа

201 – в случае успеха;

404 – в случае ошибки.

Xsd схема xml файла с запросом

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />
  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- progressType -->
  <xs:complexType name="progressType">
    <xs:sequence>
      <xs:element name="ref" minOccurs="1" maxOccurs="1" type="valueType" />
      <xs:element name="text" minOccurs="1" maxOccurs="1" type="valueType" />
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="context" type="xs:base64Binary" />
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required" />
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="error" type="errorType" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- progress -->
  <xs:element name="progress" type="progressType"/>
</xs:schema>

```

Внимание: в base64 должен быть закодирован целевой контекст маршрутизации, т.е. xml элемент типа contextType (тип определен в context.xsd).

context.xsd:

Схема xml-файла определена в разделе [Информация-о-контексте](#).


```

yIi8+CiAgICAgIDwvYWN0aw9uc4KICAgICAgPHJlc3VsdD4KICAgICAgICA8Y29udG1udWUgY29udGV4dD0iY3R4X2NpdHlf
Y29tbW9uIi8+CiAgICAgIDwvcmVzdWx0PgogICAgPC9ydWx1PgogICAgPHJ1bGUgYmFtZT0iY29sbGVjdF9leHRlcm5hbF9zd
WJzY3JpYmVycyI+CiAgICAgIDxjb25kaXRpb25zPgogICAgICAgIDxjZHBuIGRrZ210cz0iKDItNSw3LDkPz8/
Pz8iLz4KICAgICAgPC9jb25kaXRpb25zPgogICAgICA8YWN0aw9uc4KICAgICAgICA8Y2FsbGVkIHNVcm1fbmk9ImxvY2FsI
iBzb3JtX2RpZ210cz0iezEsMiwzLDQsNSw2fSivPgogICAgICAgIDxjZHBuIGRrZ210cz0iMzg1NHthYmNkZWZ9IiBuYwk9Im
5hdG1vbmFsTnVtYmVYIi8+CiAgICAgIDwvYWN0aw9uc4KICAgICAgPHJlc3VsdD4KICAgICAgICA8Y29udG1udWUgY29udGV
4dD0iY3R4X2NpdHlfY29tbW9uIi8+CiAgICAgIDwvcmVzdWx0PgogICAgPC9ydWx1PgogICAgPHJ1bGUgYmFtZT0iY29sbGVj
dF9pbmRlcm5hdG1vbmFsX3N1YnNjcmllZiXzIj4KICAgICAgPGNvbRpdG1vbnM+CiAgICAgICAgPGNkcG4gZGlnaXRzPSI4M
TAlIi8+CiAgICAgICAgPGZpbmFsIHZhbHVlPSJ0cnVlIi8+CiAgICAgIDwvY29uZG10aw9uc4KICAgICAgPGFjdG1vbnM+Ci
AgICAgICAgPGNhbGxlcCBzb3JtX25pPSJpbmRlcm5hdG1vbmFsIiBzb3JtX2RpZ210cz0ieyV9Ii8+CiAgICAgICAgPGNkcG4
gbmk9ImIudGVybW0aw9uYmVYIi8+CiAgICAgIDwvYWN0aw9uc4KICAgICAgPHJlc3VsdD4KICAgICAgICA8Y29udG1udWUgY29udGV4dD0iY3R4X2NpdHlfY29tbW9uIi8+CiAgICAgIDwvcm
VzdWx0PgogICAgPC9ydWx1PgogICAgPHJ1bGUgYmFtZT0idG9fQkFNvNFnc3ByYXYiPgogICAgICA8Y29uZG10aw9uc4KIC
AgICAgICA8Y2RwbiBkaWdpdHM9IjgXPz8iLz4KICAgICAgICA8ZmluYmVzdWU9InRydWUuLz4KICAgICAgPC9jb25kaXR
pb25zPgogICAgICA8YWN0aw9uc4KICAgICAgICA8Y2FsbGVkIHNVcm1fbmk9ImIudGVyY210eSIgc29ybV9kaWdpdHM9Ijcx
ezMnNH0iLz4KICAgICAgICA8Y2RwbiBuaT0iaW50ZXJjaXR5IiBkaWdpdHM9IntiY2R9IiBuYwk9Im5hdG1vbmFsTnVtYmVYI
i8+CiAgICAgIDwvYWN0aw9uc4KICAgICAgPHJlc3VsdD4KICAgICAgICA8Y29udG1udWUgY29udGV4dD0iY3R4X2NpdHlfY2
9tbW9uIi8+CiAgICAgIDwvcmVzdWx0PgogICAgPC9ydWx1PgogICAgPHJ1bGUgYmFtZT0idG9faw50ZXJjaXR5X2J5X29wZXI
iPgogICAgICA8Y29uZG10aw9uc4KICAgICAgICA8Y2RwbiBkaWdpdHM9Ijg1Pz8/Pz8/Pz8/
Pz8iLz4KICAgICAgICA8ZmluYmVzdWU9InRydWUuLz4KICAgICAgPC9jb25kaXRpb25zPgogICAgICA8YWN0aw9uc4KI
CAGICAgICA8Y2FsbGVkIHNVcm1fbmk9ImIudGVyY210eSIgc29ybV9kaWdpdHM9IjcxezMnNcw1LDYsNyw4LDksMTAsMTEsMT
IsMTN9Ii8+CiAgICAgICAgPGNkcG4gbmk9ImIudGVyY210eSIgZGlnaXRzPSJ7YmNkZWZnaG1qa2xtfSIgYmFtZSJuYXRpb25
hbE51bWJlciIvPgogICAgICA8L2FjdG1vbnM+CiAgICAgIDxYXN1bHQ+CiAgICAgICAgPGNvbRpbmVlIGNvbRleHQ9ImN0
eF9jaXR5X2NvbW1vbiIvPgogICAgICA8L3Jlc3VsdD4KICAgIDwvcmVzZT4KICAgIDxYdWx1IG5hbWU9ImNvbGx1Y3RfbmF0a
w9uYmVycyIvPgogICAgICA8Y29uZG10aw9uc4KICAgICAgICA8Y2RwbiBkaWdpdHM9Ijg/Pz8/Pz8/Pz8/
Ii8+CiAgICAgICAgPGZpbmFsIHZhbHVlPSJ0cnVlIi8+CiAgICAgIDwvY29uZG10aw9uc4KICAgICAgPGFjdG1vbnM+CiAgI
CAGICAgPGNhbGxlcCBzb3JtX25pPSJpbmRlcmNpdHkiIHNVcm1fZG1naXRzPSI3ezIsMyw0LDUsNiw3LDgsOSwxCwMwMX0iLz
4KICAgICAgICA8Y2RwbiBuaT0iaW50ZXJjaXR5IiBkaWdpdHM9IntiY2RlZmdoaWprfSIgYmFtZSJuYXRpb25hbE51bWJlciI
vPgogICAgICA8L2FjdG1vbnM+CiAgICAgIDxYXN1bHQ+CiAgICAgICAgPGNvbRpbmVlIGNvbRleHQ9ImN0eF9jaXR5X2Nv
bW1vbiIvPgogICAgICA8L3Jlc3VsdD4KICAgIDwvcmVzZT4KICAgIDxYdWx1IG5hbWU9ImNvbGx1Y3Rfem9uZV9zdWJzY3JpY
mVycyI+CiAgICAgIDxjb25kaXRpb25zPgogICAgICAgIDxjZHBuIGRrZ210cz0iMzg1Pz8/Pz8/
PyIvPgogICAgICA8L2NvbRpdG1vbnM+CiAgICAgIDxhY3Rpb25zPgogICAgICAgIDxjYmVzZWQgc29ybV9uaT0iaW50ZXJja
XR5IiBzb3JtX2RpZ210cz0iN3sxLDIsMyw0LDUsNiw3LDgsOSwXMH0iLz4KICAgICAgICA8Y2RwbiBuYmVYIi8+CiAgICAgIDwvYWN0aw9uc4KICAgICAgPHJlc3VsdD4KICAgICAgICA8Y29udG1udWUgY29udGV4dD0iY3R4X2N
pdHlfY29tbW9uIi8+CiAgICAgIDwvcmVzdWx0PgogICAgPC9ydWx1PgogICAgPHJ1bGUgYmFtZT0idG9faXZyIj4KICAgICAg
PGNvbRpdG1vbnM+CiAgICAgICAgPGNkcG4gZGlnaXRzPSJJPz8/
Ii8+CiAgICAgIDwvY29uZG10aw9uc4KICAgICAgPHJlc3VsdD4KICAgICAgICA8Y29udG1udWUgY29udGV4dD0iY3R4X3RvX
212ciIvPgogICAgICA8L3Jlc3VsdD4KICAgIDwvcmVzZT4KICAgL2NvbRleHQ+Cg==</context>
</request>
</in>

```

Данная base64 строка соответствует контексту:

```

<?xml version="1.0" encoding="UTF-8"?>
  <context xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ecss_routing.xsd" name="ctx_from_local" np="np_240" digitmap="auto">
    <rule name="smg-5">
      <conditions>
        <cdpn digits="2408??" />
        <final value="true" />
      </conditions>
      <actions>
        <called sorm_ni="local" sorm_digits="{1,2,3,4,5,6}" />
        <cdpn digits="3854{abcdef}" nai="nationalNumber" />
      </actions>
      <result>
        <continue context="ctx_city_common" />
      </result>
    </rule>
    <rule name="sorm_test">
      <conditions>
        <cdpn digits="240???" />
        <final value="true" />
      </conditions>
      <actions>
        <called sorm_ni="private" sorm_digits="{1,2,3,4,5,6}" />
        <cdpn ni="local" nai="subscriberNumber" />
      </actions>
      <result>
        <local />
      </result>
    </rule>
    <rule name="normalize_subscriber_to_national">
      <conditions>
        <cgpn digits="240???" nai="subscriberNumber" />
      </conditions>
      <actions>
        <calling sorm_ni="private" sorm_digits="73854{1,2,3,4,5,6}" />
        <cgpn digits="3854240{def}" nai="nationalNumber" />
      </actions>
      <result>
        <continue />
      </result>
    </rule>
    <rule name="to_city_emergency_0x">
      <conditions>
        <cdpn digits="0(1,2,3,4,7,9)" />
        <final value="true" />
      </conditions>
      <actions>
        <called sorm_ni="emergency" sorm_digits="{1,2}" />
        <cdpn ni="emergency" nai="subscriberNumber" />
      </actions>
    </rule>
  </context>

```

```

</actions>
<result>
  <continue context="ctx_to_ewsd"/>
</result>
</rule>
<rule name="to_city_emergency_00x">
  <conditions>
    <cdpn digits="00(2,4)"/>
    <final value="true"/>
  </conditions>
  <actions>
    <called sorm_ni="emergency" sorm_digits="{1,2,3}"/>
    <cdpn ni="emergency" nai="subscriberNumber"/>
  </actions>
  <result>
    <continue context="ctx_to_ewsd"/>
  </result>
</rule>
<rule name="to_city_emergency_xxx">
  <conditions>
    <cdpn digits="(05?,06?,1??)"/>
    <final value="true"/>
  </conditions>
  <actions>
    <called sorm_ni="emergency" sorm_digits="{1,2,3}"/>
    <cdpn ni="emergency" nai="subscriberNumber"/>
  </actions>
  <result>
    <continue context="ctx_to_ewsd"/>
  </result>
</rule>
<rule name="to_emerg_8800">
  <conditions>
    <cdpn digits="8800?????"/>
    <final value="true"/>
  </conditions>
  <actions>
    <called sorm_ni="emergency" sorm_digits="7{2,3,4,5,6,7,8,9,10,11}"/>
    <cdpn ni="emergency" digits="{bcdefghijk}" nai="nationalNumber"/>
  </actions>
  <result>
    <continue context="ctx_city_common"/>
  </result>
</rule>
<rule name="collect_external_subscribers">
  <conditions>
    <cdpn digits="(2-5,7,9)????"/>
  </conditions>
  <actions>
    <called sorm_ni="local" sorm_digits="{1,2,3,4,5,6}"/>

```

```

    <cdpn digits="3854{abcdef}" nai="nationalNumber"/>
  </actions>
  <result>
    <continue context="ctx_city_common"/>
  </result>
</rule>
<rule name="collect_international_subscribers">
  <conditions>
    <cdpn digits="810%"/>
    <final value="true"/>
  </conditions>
  <actions>
    <called sorm_ni="international" sorm_digits="{%}" />
    <cdpn ni="international" digits="{%}" nai="internationalNumber"/>
  </actions>
  <result>
    <continue context="ctx_city_common"/>
  </result>
</rule>
<rule name="to_BAMTS_sprav">
  <conditions>
    <cdpn digits="81??"/>
    <final value="true"/>
  </conditions>
  <actions>
    <called sorm_ni="intercity" sorm_digits="71{3,4}" />
    <cdpn ni="intercity" digits="{bcd}" nai="nationalNumber"/>
  </actions>
  <result>
    <continue context="ctx_city_common"/>
  </result>
</rule>
<rule name="to_intercity_by_oper">
  <conditions>
    <cdpn digits="85?????????"/>
    <final value="true"/>
  </conditions>
  <actions>
    <called sorm_ni="intercity" sorm_digits="75{3,4,5,6,7,8,9,10,11,12,13}" />
    <cdpn ni="intercity" digits="{bcdefghijklm}" nai="nationalNumber"/>
  </actions>
  <result>
    <continue context="ctx_city_common"/>
  </result>
</rule>
<rule name="collect_national_subscribers">
  <conditions>
    <cdpn digits="8?????????"/>
    <final value="true"/>
  </conditions>

```

```

<actions>
  <called sorm_ni="intercity" sorm_digits="7{2,3,4,5,6,7,8,9,10,11}"/>
  <cdpn ni="intercity" digits="{bcdefghijk}" nai="nationalNumber"/>
</actions>
<result>
  <continue context="ctx_city_common"/>
</result>
</rule>
<rule name="collect_zone_subscribers">
  <conditions>
    <cdpn digits="385??????"/>
  </conditions>
  <actions>
    <called sorm_ni="intercity" sorm_digits="7{1,2,3,4,5,6,7,8,9,10}"/>
    <cdpn nai="nationalNumber"/>
  </actions>
  <result>
    <continue context="ctx_city_common"/>
  </result>
</rule>
<rule name="to_ivr">
  <conditions>
    <cdpn digits="C???" />
  </conditions>
  <result>
    <continue context="ctx_to_ivr"/>
  </result>
</rule>
</context>

```

Hc_cluster_statistics - Управление системой сбора статистики для кластера

- [Hc_cluster_statistics_add](#)
- [Hc_cluster_statistics_delete](#)
- [Hc_cluster_statistics_statmodinfo](#)
- [Hc_cluster_statistics_addcolmap](#)
- [Hc_cluster_statistics_delcolmap](#)
- [Hc_cluster_statistics_list](#)

Hc_cluster_statistics_add

Команда позволяет добавить новую задачу по сбору статистики для кластера.

Аналог команды в CoCon:

/cluster/mediator/<md>/statistics/add

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cluster_statistics_add

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name = "moduleType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="e502Type1"/>
      <xs:enumeration value="e502Type10"/>
      <xs:enumeration value="e502Type11"/>
      <xs:enumeration value="e502Type12"/>
      <xs:enumeration value="e502Type15"/>
      <xs:enumeration value="e502Type16"/>
      <xs:enumeration value="e502Type2"/>
      <xs:enumeration value="e502Type20"/>
      <xs:enumeration value="e502Type3"/>
      <xs:enumeration value="e502Type4"/>
      <xs:enumeration value="e502Type5"/>
      <xs:enumeration value="e502Type6"/>
      <xs:enumeration value="e502Type9"/>
      <xs:enumeration value="hostResources"/>
      <xs:enumeration value="mgTraffic"/>
      <xs:enumeration value="q752Table1"/>
      <xs:enumeration value="q752Table15"/>
      <xs:enumeration value="q752Table2"/>
      <xs:enumeration value="q752Table3"/>
      <xs:enumeration value="q752Table4"/>
      <xs:enumeration value="q752Table5"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name = "exportNameType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="txt_file_export">
        <xs:annotation>
          <xs:documentation xml:lang="en">
            For txt_file_export should be set 'stat' property.
          </xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="rpss_db_export">
        <xs:annotation>
          <xs:documentation xml:lang="en">
            For rpss_db_export should be set 'connStringName', 'recoveryDir'
properties.
          </xs:documentation>
        </xs:annotation>
      </xs:enumeration>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="exportType">

```

```

    <xs:attribute name="name" type="xs:string" use="required" />
    <xs:attribute name="value" type="xs:string" use="required" />
</xs:complexType>

<xs:complexType name="filterType">
  <xs:attribute name="name" type="xs:string" use="required" />
</xs:complexType>

<xs:complexType name="clusterType">
  <xs:sequence>
    <xs:element name="export" type="exportType" minOccurs="0" maxOccurs="unbounded" />
    <xs:element name="filter" type="filterType" minOccurs="0" maxOccurs="unbounded" />
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="module" type="moduleType" use="required"/>
  <xs:attribute name="interval" type="xs:integer" use="required"/>
  <xs:attribute name="export" type="exportNameType" use="required"/>
</xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="cluster" type="clusterType" minOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

Запрос: http://192.168.1.21:9999/commands/cluster_statistics_add

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cluster_statistics_add.xsd">
  <cluster name="md1" module="e502Type1" interval="15" export="txt_file_export">
    <export name="stat" value="/var/lib/ecss/statistics" />
    <filter name="biysk.local" />
  </cluster>
</in>

```

Ответ:201

Пример 2: (запись статистики в БД)

Запрос: http://192.168.1.21:9999/commands/cluster_statistics_add


```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cluster_statistics_add.xsd">
  <cluster name="md1" module="e502Type1" interval="10" export="rpss_db_export">
    <export name="connStringName" value="stat_db" />
    <export name="recoveryDir" value="/tmp" />
    <filter name="biysk.local" />
  </cluster>
</in>
```

Ответ:201

[Hc_cluster_statistics_delete](#)

Команда позволяет удалить задачу по сбору статистики в кластере.

Аналог команды в CoCon:

/cluster/mediator/<md>/statistics/delete

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса: http://host:port/commands/cluster_statistics_delete

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="clusterType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="cluster" type="clusterType" minOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: `http://192.168.1.21:9999/commands/cluster_statistics_delete`

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_statistics_delete.xsd">
  <cluster name="md1" id="65656461-6262-6236-3739-633230303936" />
</in>
```

Ответ:204

[Hc_cluster_statistics_statmodinfo](#)

Команда позволяет просмотреть информации по маппированию колонок в кластере.

Аналог команды в CoCon:

`/cluster/mediator/<md>/statistics/statmodinfo <ModuleName> [-m]`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/cluster_statistics_statmodinfo`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="clusterType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="module" type="xs:string" use="required"/>
    <xs:attribute name="flag" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="column1Type">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="xs:string" use="required"/>
    <xs:attribute name="constrait" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="columnsType">
    <xs:sequence>
      <xs:element name="column" type="column1Type" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="column2Type">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="new_name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="columns2Type">
    <xs:sequence>
      <xs:element name="column" type="column2Type" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="cluster" type="clusterType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="columns" type="columnsType" />
        <xs:element name="m_columns" type="columns2Type" />
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/cluster_statistics_statmodinfo

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cluster_statistics_statmodinfo.xsd">
  <cluster name="md1" module="e502Type10" flag="-m"/>
</in>
```

Ответ:

200 ok

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cluster_statistics_statmodinfo.xsd">
  <m_columns/>
</out>
```

Запрос: http://192.168.1.21:9999/commands/cluster_statistics_statmodinfo

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cluster_statistics_statmodinfo.xsd">
  <cluster module="e502Type1" name="md1" />
</in>
```

Ответ:

200 ok

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cluster_statistics_statmodinfo.xsd">
  <columns description="Overall measurements on originating traffic">
    <column name="seizures" type="integer" constraint="undefined" description="Number of
originating seizures"/>
    <column name="notRouted" type="integer" constraint="undefined" description="Number of call
attempts not routed due to: no dialling, incomplete dialling, invalid address"/>
    <column name="congestion" type="integer" constraint="undefined" description="Number of
call attempts lost due to internal congestion"/>
  </columns>
</out>
```

[Hc_cluster_statistics_addcolmap](#)

Команда позволяет добавить маппирования колонок в кластере.

Аналог команды в CoCon:

/cluster/mediator/<md>/statistics/addcolmap

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cluster_statistics_addcolmap

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить двумя сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name = "moduleType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="e502Type1"/>
      <xs:enumeration value="e502Type10"/>
      <xs:enumeration value="e502Type11"/>
      <xs:enumeration value="e502Type12"/>
      <xs:enumeration value="e502Type15"/>
      <xs:enumeration value="e502Type16"/>
      <xs:enumeration value="e502Type2"/>
      <xs:enumeration value="e502Type20"/>
      <xs:enumeration value="e502Type3"/>
      <xs:enumeration value="e502Type4"/>
      <xs:enumeration value="e502Type5"/>
      <xs:enumeration value="e502Type6"/>
      <xs:enumeration value="e502Type9"/>
      <xs:enumeration value="hostResources"/>
      <xs:enumeration value="mgTraffic"/>
      <xs:enumeration value="q752Table1"/>
      <xs:enumeration value="q752Table15"/>
      <xs:enumeration value="q752Table2"/>
      <xs:enumeration value="q752Table3"/>
      <xs:enumeration value="q752Table4"/>
      <xs:enumeration value="q752Table5"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="clusterType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="module" type="moduleType" use="required"/>
    <xs:attribute name="column" type="xs:string" use="required"/>
    <xs:attribute name="new_column" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="cluster" type="clusterType" minOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/cluster_statistics_addcolmap

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="cluster_statistics_addcolmap.xsd">
  <cluster name="md1" module="e502Type1" column="notRouted" new_column="route_error" />
</in>
```

Ответ:

201

[Hc_cluster_statistics_delcolmap](#)

Команда позволяет удалить маппирования колонок в кластере.

Аналог команды в CoCon:

/cluster/mediator/<md>/statistics/delcolmap

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cluster_statistics_delcolmap

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="clusterType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="module" type="xs:string" use="required"/>
    <xs:attribute name="column" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="cluster" type="clusterType" minOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/cluster_statistics_delcolmap

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="cluster_statistics_delcolmap.xsd">
  <cluster name="md1" module="e502Type1" column="notRouted"/>
</in>
```

Ответ:204

[Hc_cluster_statistics_list](#)

Команда позволяет получить список задач по сбору статистики в кластере.

Аналог команды в CoCon:

`/cluster/mediator/<md>/statistics/list`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cluster_statistics_list

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="clusterType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:simpleType name = "moduleType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="e502Type1"/>
      <xs:enumeration value="e502Type10"/>
      <xs:enumeration value="e502Type11"/>
      <xs:enumeration value="e502Type12"/>
      <xs:enumeration value="e502Type15"/>
      <xs:enumeration value="e502Type16"/>
      <xs:enumeration value="e502Type2"/>
      <xs:enumeration value="e502Type20"/>
      <xs:enumeration value="e502Type3"/>
      <xs:enumeration value="e502Type4"/>
      <xs:enumeration value="e502Type5"/>
      <xs:enumeration value="e502Type6"/>
      <xs:enumeration value="e502Type9"/>
      <xs:enumeration value="hostResources"/>
      <xs:enumeration value="mgTraffic"/>
      <xs:enumeration value="q752Table1"/>
      <xs:enumeration value="q752Table15"/>
      <xs:enumeration value="q752Table2"/>
      <xs:enumeration value="q752Table3"/>
      <xs:enumeration value="q752Table4"/>
      <xs:enumeration value="q752Table5"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="idType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="module" type="moduleType" use="required"/>
    <xs:attribute name="interval" type="xs:integer" use="required"/>
    <xs:attribute name="start_time" type="xs:string" use="required"/>
    <xs:attribute name="end_time" type="xs:string" use="required"/>
    <xs:attribute name="exports" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="cluster" type="clusterType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="id" type="idType" minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/cluster_statistics_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="cluster_group_list.xsd">
  <cluster name="md1" />
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="cluster_statistics_list.xsd">
  <id name="30363235-3262-3231-3163-333732343163" module="e502Type1" interval="900"
start_time="2019/11/08 08:30:57" end_time="undefined" exports="txt_file_export"/>
</out>

```

Hc_cluster - Управление кластерами

- [Hc_cluster_info](#)
- [Hc_list_of_adapter_clusters](#)
- [Hc_get_cluster_properties](#)
- [Hc_cluster_set](#)
- [Hc_cluster_clear](#)
- [Hc_clusters_list](#)
- [Hc_cluster_properties](#)

Hc_cluster_info

Команда позволяет вывести информацию о кластере.

Аналог команды в CoCon:

`/cluster/storage/<CLUSTER>/cluster/<CLUSTER_TYPE>/<CLUSTER_NAME>/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cluster_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="common.xsd"/>
  <!-- nodeType -->
  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- propertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- clusterType -->
  <xs:complexType name="clusterType">
    <xs:all>
      <xs:element name="nodes">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="node" type="nodeType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="properties">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:all>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="role" type="roleType" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="cluster">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            <xs:attribute name="role" type="roleType" use="required"/>
            <xs:attribute name="name" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="cluster" type="clusterType" minOccurs="1" maxOccurs="1"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

common.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- dateTimeType -->
  <xs:simpleType name="dateTimeType">
    <xs:restriction base="xs:string">
      <xs:pattern value="\d{4}/\d{2}/\d{2} \d{2}:\d{2}:\d{2}(\.\d{0,3})?"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- roleType -->
  <xs:simpleType name="roleType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="core"/>
      <xs:enumeration value="adapter"/>
      <xs:enumeration value="storage"/>
      <xs:enumeration value="mediator"/>
      <xs:enumeration value="tts"/>
      <xs:enumeration value="bus"/>
      <xs:enumeration value="all"/>
    </xs:restriction>
  </xs:simpleType>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/cluster_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cluster_info.xsd">
  <cluster name="sip1" role="adapter" storage="ds1" />
</in>

```

Ответ:200


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- adapterType -->
  <xs:simpleType name="adapterType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="sip"/>
      <xs:enumeration value="megaco"/>
      <xs:enumeration value="all"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="type" type="adapterType" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="adapters">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="adapter" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="name" type="xs:string" use="required"/>
                  <xs:attribute name="type" type="adapterType" use="required"/>
                  <xs:attribute name="active" type="xs:boolean" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/list_of_adapter_clusters

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_list_of_adapter_clusters.xsd">
  <request type="all" />
</in>
```

ОТВЕТ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_list_of_adapter_clusters.xsd">
  <adapters>
    <adapter name="sip1" type="sip" active="true"/>
    <adapter name="megaco1" type="megaco" active="true"/>
  </adapters>
</out>
```

[Hc_get_cluster_properties](#)

Команда позволяет вывести свойства кластера. Команда возвращает информацию, что и команда "Информация о кластере", только в другом формате.

Аналог команды в CoCon:

`/cluster/storage/<CLUSTER>/cluster/<CLUSTER_TYPE>/<CLUSTER_NAME>/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса/ответа:

http://host:port/commands/get_cluster_properties

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="common.xsd"/>
  <!-- nodeType -->
  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- propertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- clusterType -->
  <xs:complexType name="clusterType">
    <xs:all>
      <xs:element name="properties">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:all>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="role" type="roleType" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="cluster">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            <xs:attribute name="role" type="roleType" use="required"/>
            <xs:attribute name="name" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="cluster" type="clusterType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/get_cluster_properties

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="cluster_properties.xsd">
  <cluster name="ds1" role="storage" storage="ds1" />
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="cluster_properties.xsd">
  <cluster name="ds1" role="storage">
    <properties>
      <property name="nodes" value="[ds1@ecss1,ds1@ecss2]"/>
      <property name="isActive" value="true"/>
    </properties>
  </cluster>
</out>
```

hc_cluster_set

Команда используется для задания свойств кластера.

Аналог команды в CoCon:

```
/cluster/storage/<CLUSTER>/cluster/<CLUSTER_TYPE>/<CLUSTER_NAME>/set
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cluster_set

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими двумя сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="common.xsd"/>
  <!-- propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="cluster" minOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded" />
            </xs:sequence>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            <xs:attribute name="role" type="roleType" use="required"/>
            <xs:attribute name="name" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:Запрос: http://192.168.1.21:9999/commands/cluster_set

Параметры:

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_cluster_set.xsd">
  <cluster role="storage" storage="ds1" name="ds1">
    <property name="key1" value="value1"/>
    <property name="ilist" value="[1,2,3]"/>
    <property name="clist" value="[1,2,3, [a, b], test]"/>
  </cluster>
</in>

```

Ответ 403 - команда доступна с правами root:

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="common_error.xsd">
  <error cmd="cluster_set" reason="no_rights" body="{not_authorized,no_rights}"/>
</out>
```

Нс_cluster_clear

Команда позволяет удалить свойство кластера (одного или все).

Аналог команды в CoCon:

```
/cluster/storage/<CLUSTER>/cluster/<CLUSTER_TYPE>/<CLUSTER_NAME>/clean
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cluster_clear

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 403,404 – в случае ошибки.

XSD-схема XML-файла с запросом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="common.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="cluster">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="property" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="name" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            <xs:attribute name="name" type="xs:string" use="required"/>
            <xs:attribute name="role" type="roleType" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

⚠ Если не одно имя свойства не задано, то будут удалены все свойства данного кластера.

Пример:

Запрос: http://192.168.1.21:9999/commands/cluster_clear

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_cluster_clear.xsd">
  <cluster role="core" storage="ds1" name="core1">
    <property name="a"/>
    <property name="b"/>
  </cluster>
</in>

```

Ответ: 403 (только с root-правами)

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="common_error.xsd">
  <error cmd="cluster_clear" reason="no_rights" body="{not_authorized,no_rights}"/>
</out>
```

Нс_clusters_list

Команда возвращает список кластеров, определенных в системе.

Аналог команды в CoCon:

/cluster/storage/<CLUSTER>/cluster/list

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/list_of_clusters

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="common.xsd"/>
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="role" type="roleType" use="optional"/>
            <xs:attribute name="storage" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="clusters">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="cluster" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="name" type="xs:string" use="required"/>
                  <xs:attribute name="role" type="roleType" use="required"/>
                  <xs:attribute name="active" type="xs:boolean" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/list_of_clusters

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_list_of_clusters.xsd">
  <request storage="ds1"/>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_list_of_clusters.xsd">
  <clusters>
    <cluster name=".virtual" role="adapter" active="true"/>
    <cluster name="sip1" role="adapter" active="true"/>
    <cluster name="core1" role="core" active="true"/>
    <cluster name="ds1" role="storage" active="true"/>
    <cluster name="megaco1" role="adapter" active="true"/>
    <cluster name="md1" role="mediator" active="true"/>
    <cluster name=".system_bridge" role="adapter" active="true"/>
  </clusters>
</out>

```

Hc_cluster_properties

Команда возвращает свойства кластера. Hc_cluster_properties возвращает ту же информацию, что и команда [Hc_cluster_info](#), только в другом формате.

Аналог команды в CoCon:

/cluster/storage/<CLUSTER>/cluster/<CLUSTER_TYPE>/<CLUSTER_NAME>/info

Метод http запроса:

POST

Шаблон http запроса/ответа:

http://host:port/commands/get_cluster_properties

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема xml файла с запросом/ответом:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="common.xsd"/>
  <!-- nodeType -->
  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- propertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- clusterType -->
  <xs:complexType name="clusterType">
    <xs:all>
      <xs:element name="properties">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:all>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="role" type="roleType" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="cluster">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            <xs:attribute name="role" type="roleType" use="required"/>
            <xs:attribute name="name" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="cluster" type="clusterType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```
</xs:schema
```

common.xsd:

См. страницу [Hc_cluster_info](#).

Пример:

Запрос: http://192.168.1.21:9999/commands/get_cluster_properties

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="cluster_properties.xsd">
  <cluster name="sip1" role="adapter" storage="ds1" />
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cluster_properties.xsd">
  <cluster name="sip1" role="adapter">
    <properties>
      <property name="nodes" value="[sip1@ecss1,sip1@ecss2]"/>
      <property name="isActive" value="true"/>
    </properties>
  </cluster>
</out>
```

 12.11.2019

Hc_cn_options - Управление службой нотификации вызова CN

- [Hc_cn_options_clean](#)
- [Hc_cn_options_info](#)
- [Hc_cn_options_set](#)

Hc_cn_options_clean

Команда сброса настроек службы CN.

Аналог команды в CoCon:

```
/cluster/core/<CORE>/cn/clean
```

```
/domain/<DOMAIN>/cn/clean
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cn_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean properties of call notifier service.
      Usage: http://server:9999/commands/cn_options_clean.
      Implementation: hc_cn_options_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/cn_options_clean

```
<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cn_options_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="jabber_release_template"/>
        <option name="email_release_template"/>
        <option name="release_call"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cn_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="email_release_template"/>
        <success option="jabber_release_template"/>
        <success option="release_call"/>
      </domain>
    </domain_options>
  </result>
</out>
```

[Hc_cn_options_info](#)

Команда просмотра установленных значений параметров службы CN.

Аналог команды в CoCon:

`/cluster/core/<CORE>/cn/info`

/domain/<DOMAIN>/cn/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cn_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of call notifier service.
      Usage: http://server:9999/commands/cn_options_info.
      Implementation: hc_cn_options_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="cnOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="jabber_incoming_template" type="xs:string" use="optional"/>
        <xs:attribute name="jabber_release_template" type="xs:string" use="optional"/>
        <xs:attribute name="jabber_incoming_plain_template" type="xs:string"
use="optional"/>
        <xs:attribute name="jabber_release_plain_template" type="xs:string"
use="optional"/>
        <xs:attribute name="release_call" type="booleanType" use="optional"/>
        <xs:attribute name="email_release_template" type="xs:string" use="optional"/>
        <xs:attribute name="incoming_call" type="booleanType" use="optional"/>
        <xs:attribute name="email_from_name" type="xs:string" use="optional"/>
        <xs:attribute name="email_subject_format" type="xs:string" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainInfoInType" minOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```
        </xs:all>
    </xs:complexType>
    </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>
```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cn_options_info

```
<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_cn_options_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>
```

Код ответа:

200

Ответ:


```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cn_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="cnOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="cnOptionsType" release_call="false"
jabber_release_template="%DISPLAYNAME% (<a href='%LINK_CALL_TO%=%CALLERID%'>%CALLERID%</a>)
called you (%CALLED%) at %TIME% %DATE%" jabber_release_plain_template="%DISPLAYNAME% (%CALLERID%
- call back %LINK_CALL_TO%=%CALLERID% ) called you (%CALLED%) at %TIME% %DATE%"
jabber_incoming_template="%DISPLAYNAME% (<a href='%LINK_CALL_TO%=%CALLERID%'>%CALLERID%</a>)
calling you (%CALLED%) at %TIME% %DATE%" jabber_incoming_plain_template="%DISPLAYNAME%
(%CALLERID% - call back %LINK_CALL_TO%=%CALLERID% ) calling you (%CALLED%) at %TIME% %DATE%"
incoming_call="true" email_subject_format="%DISPLAYNAME% (%CALLERID%) called you (%CALLED%) at
%TIME% %DATE%" email_release_template="%DISPLAYNAME% (%CALLERID% - call back
%LINK_CALL_TO%=%CALLERID% ) called you (%CALLED%) at %TIME% %DATE%" email_from_name="ECSS-NO-
REPLY"/>
      </domain>
    </domain_options>
  </result>
</out>

```

hc_cn_options_set

Команда изменения значений параметров службы CN.

Аналог команды в CoCon:

/cluster/core/<CORE>/cn/set

/domain/<DOMAIN>/cn/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cn_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set call notifier service properties
      Usage: http://server:9999/commands/cn_options_set.
      Implementation: hc_cn_options_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="cnOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="jabber_incoming_template" type="xs:string" use="optional"/>
        <xs:attribute name="jabber_release_template" type="xs:string" use="optional"/>
        <xs:attribute name="jabber_incoming_plain_template" type="xs:string"
use="optional"/>
        <xs:attribute name="jabber_release_plain_template" type="xs:string"
use="optional"/>
        <xs:attribute name="release_call" type="booleanType" use="optional"/>
        <xs:attribute name="email_release_template" type="xs:string" use="optional"/>
        <xs:attribute name="incoming_call" type="booleanType" use="optional"/>
        <xs:attribute name="email_from_name" type="xs:string" use="optional"/>
        <xs:attribute name="email_subject_format" type="xs:string" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
            <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/cn_options_set

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cn_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="cnOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="cnOptionsType" incoming_call="true" release_call="false"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_cn_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="incoming_call"/>
        <success option="release_call"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_cocon_configure - Управление CoCon-сессией

- [Нс_cocon_configure_info](#)
- [Нс_cocon_configure_start](#)
- [Нс_cocon_configure_stop](#)

Нс_cocon_configure_info

Команда позволяет получить список всех блокировок CoCon-сессий.

Аналог команды в CoCon:

cocon/configure-info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cocon_configure_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="coconLockType">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="system" type="xs:string" use="required"/>
    <xs:attribute name="locks" type="xs:string" use="required"/>
    <xs:attribute name="locked_by" type="xs:string" use="required"/>
    <xs:attribute name="deadline" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType/>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="lock" type="coconLockType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/cocon_configure_info

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="cocon_configure_info.xsd">
  <request />
</in>

```

Код ответа:

200

Ответ(время в UTC):

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_configure_info.xsd">
  <response>
    <lock deadline="17.11.2019 03:58:45" locked_by="admin" locks="ct.office" system="ECSS-10-
COCON" id="{elm_token,<<"0625eb10958e807a">>,<<"ECSS-10-COCON">>}"/>
  </response>
</out>
```

Нс_cocon_configure_start

Команда позволяет перевести CoCon-сессии в режим единоличного конфигурирования (домена или всей системы).

Аналог команды в CoCon:

cocon/configure-start

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cocon_configure_start

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- okType -->
  <xs:complexType name="okType">
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!-- ok -->
        <xs:element name="ok" type="okType" minOccurs="0"/>
        <!-- error -->
        <xs:element name="error" type="errorType" minOccurs="0"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema

```

Пример 1:

Запрос:

http://192.168.1.21:9999/commands/cocon_configure_start

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_configure_start.xsd">
  <request domain="ct.office" />
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_configure_start.xsd">
  <ok/>
</out>
```

Нс_cocon_configure_stop

Команда позволяет вывести CoCon-сессию из режима единоличного конфигурирования (домена или всей системы).

Аналог команды в CoCon:

cocon/configure-stop

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cocon_configure_stop

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- okType -->
  <xs:complexType name="okType">
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!-- ok -->
        <xs:element name="ok" type="okType" minOccurs="0"/>
        <!-- error -->
        <xs:element name="error" type="errorType" minOccurs="0"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

Запрос:

http://192.168.1.21:9999/commands/cocon_configure_stop

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_configure_stop.xsd">
  <request domain="biysk.local" />
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="cocon_configure_stop.xsd">
  <ok/>
</out>
```

Пример 2:

Запрос:

http://192.168.1.21:9999/commands/cocon_configure_stop

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="cocon_configure_stop.xsd">
  <request/>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="cocon_configure_stop.xsd">
  <error cmd="hc_cocon_configure_stop" reason="lock_not_found" body="lock_not_found"/>
</out>
```

Нс_cocon - Управление пользователями системы

- [Hc_cocon_add_user](#)
- [Hc_cocon_del_user](#)
- [Hc_cocon_add_user_to_group](#)
- [Hc_cocon_del_user_from_group](#)
- [Hc_my_groups](#)
- [Hc_cocon_passwd](#)
- [Hc_cocon_check_passwd](#)
- [Hc_cocon_users_list](#)
- [Hc_cocon_groups_list](#)
- [Hc_cocon_roles_list](#)
- [Hc_cocon_user_property_clean](#)
- [Hc_cocon_user_property_info](#)
- [Hc_cocon_user_property_set](#)

Ис_cocon_add_user

Команда добавляет нового пользователя в систему, позволяет добавить его в группы пользователей/администраторов системы ECSS-10, определенных виртуальных АТС (доменов).

Аналог команды в CoCon:

`cocon/add-user`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/cocon_add_user`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- okType-->
  <xs:complexType name="okType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- warningType-->
  <xs:complexType name="warningType">
    <xs:all>
      <xs:element name="error_groups">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="group" type="warningGroupType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:all>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- weakPasswordType-->
  <xs:complexType name="weakPasswordType">
    <xs:all>
      <xs:element name="failed_restrictions" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="restriction" type="restrictionType" minOccurs="1"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
  <!-- domainType-->
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- warningGroupType-->
  <xs:complexType name="warningGroupType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="cause" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- restrictionType-->
  <xs:complexType name="restrictionType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="args" type="xs:string" use="required"/>
    <xs:attribute name="msg" type="xs:string" use="required"/>
  </xs:complexType>

```

```

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="user">
        <xs:complexType>
          <xs:all>
            <xs:element name="udomains">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="domain" type="domainType" minOccurs="0"
maxOccurs="unbounded"/>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
            <xs:element name="adomains">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="domain" type="domainType" minOccurs="0"
maxOccurs="unbounded"/>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
          </xs:all>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="password" type="xs:string" use="required"/>
          <xs:attribute name="is_user" type="xs:boolean" default="false"/>
          <xs:attribute name="is_admin" type="xs:boolean" default="false"/>
          <xs:attribute name="is_password_expired" type="xs:boolean" default="false"
"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- out-->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <!-- ok -->
      <xs:element name="ok" type="okType" minOccurs="0"/>
      <!-- already_exists -->
      <xs:element name="already_exists" type="okType" minOccurs="0"/>
      <!-- weak_password -->
      <xs:element name="weak_password" type="weakPasswordType" minOccurs="0"/>
      <!-- warning -->
      <xs:element name="warning" type="warningType" minOccurs="0"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

```

```
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/cocon_add_user

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_add_user.xsd">
  <user is_user="true" password="test123" is_admin="false" name="test_user">
    <udomains>
      <domain name="biysk.local"/>
      <domain name="ct.office"/>
    </udomains>
    <adomains>
      <domain name="ct.office"/>
    </adomains>
  </user>
</in>
```

Ответ:

200. Если пользователь с таким именем уже существует - выдается ошибка 404

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_add_user.xsd">
  <ok name="test_user"/>
</out>
```

Если пароль не соответствует требованиям безопасности:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_add_user.xsd">
  <weak_password>
    <failed_restrictions>
      <restriction name="min_length" args="8" msg="Passwords must be at least 8 characters
in length"/>
    </failed_restrictions>
  </weak_password>
</out>
```

Нс_cocon_del_user

Команда позволяет удалить пользователя из системы.

Аналог команды в CoCon:

`cocon/del-user`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/cocon_del_user`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- errorType-->
  <xs:complexType name="errorType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="user">
          <xs:complexType>
            <xs:attribute name="name" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!-- error -->
        <xs:element name="error" type="errorType" minOccurs="0"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/cocon_del_user

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_del_user.xsd">
  <user name="test_sveta"/>
</in>

```

Ответ:

204

Запрос на удаление несуществующего пользователя


```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_del_user.xsd">
  <user name="test-sveta"/>
</in>
```

Ответ:

404

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="common_error.xsd">
  <error cmd="cocon_del_user" reason="error" body="<?xml version="1.0"?><out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="cocon_del_user.xsd"><error name="undefined" reason="Error:
    {undefined_state,exists}"/></out>" />
  </out>
```

[Hc_cocon_add_user_to_group](#)

Команда добавляет пользователя в указанную группу доступа.

Аналог команды в CoCon:

cocon/add-user-to-group

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cocon_add_user_to_group

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- okType-->
  <xs:complexType name="okType">
    <xs:attribute name="user" type="xs:string" use="required"/>
    <xs:attribute name="group" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- errorType-->
  <xs:complexType name="errorType">
    <xs:attribute name="user" type="xs:string" use="required"/>
    <xs:attribute name="group" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- groupType-->
  <xs:complexType name="groupType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- groupsType-->
  <xs:complexType name="groupsType">
    <xs:sequence>
      <xs:element name="group" type="groupType" minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- userType-->
  <xs:complexType name="userType">
    <xs:all>
      <xs:element name="groups" type="groupsType"/>
    </xs:all>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:sequence>
      <!-- ok -->
      <xs:element name="ok" type="okType" minOccurs="0" maxOccurs="unbounded"/>
      <!-- error -->
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="user" type="userType" minOccurs="1" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Примеры:

Запрос: http://192.168.1.21:9999/commands/cocon_add_user_to_group

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_add_user_to_group.xsd">
  <user name="test_user">
    <groups>
      <group name="ecss-admin"/>
      <group name="ecss-user"/>
    </groups>
  </user>
  <user name="test2">
    <groups>
      <group name="ecss-biysk.local-domain-user"/>
      <group name="ecss-user"/>
      <group name="ecss-biysk.local-domain-admin"/>
    </groups>
  </user>
</in>

```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_add_user_to_group.xsd">
  <result>
    <ok user="test_user" group="ecss-admin"/>
    <ok user="test_user" group="ecss-user"/>
    <ok user="test2" group="ecss-biysk.local-domain-user"/>
    <ok user="test2" group="ecss-user"/>
    <ok user="test2" group="ecss-biysk.local-domain-admin"/>
  </result>
</out>
```

Запрос на добавление пользователя в несуществующую группу:

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_add_user_to_group.xsd">
  <user name="txtuser">
    <groups>
      <group name="ecss-ssw"/>
      <group name="ecss-user"/>
    </groups>
  </user>
</in>
```

Ответ:

404

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="common_error.xsd">
  <error cmd="cocon_add_user_to_group" reason="error" body="{error,"Bad result:
  {undefined_state,noSuchObject}"}"/>
</out>
```

[Hc_cocon_del_user_from_group](#)

Команда позволяет удалить пользователя из указанной группы доступа.

Аналог команды в CoCon:

cocon/del-user-from-group

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cocon_del_user_from_group

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить двумя сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- okType-->
  <xs:complexType name="okType">
    <xs:attribute name="user" type="xs:string" use="required"/>
    <xs:attribute name="group" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- errorType-->
  <xs:complexType name="errorType">
    <xs:attribute name="user" type="xs:string" use="required"/>
    <xs:attribute name="group" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- groupType-->
  <xs:complexType name="groupType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- groupsType-->
  <xs:complexType name="groupsType">
    <xs:sequence>
      <xs:element name="group" type="groupType" minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- userType-->
  <xs:complexType name="userType">
    <xs:all>
      <xs:element name="groups" type="groupsType"/>
    </xs:all>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:sequence>
      <!-- ok -->
      <xs:element name="ok" type="okType" minOccurs="0" maxOccurs="unbounded"/>
      <!-- error -->
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="user" type="userType" minOccurs="1" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out-->

```

```

<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/cocon_del_user_from_group

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_del_user_to_group.xsd">
  <user name="test_user">
    <groups>
      <group name="ecss-admin"/>
    </groups>
  </user>
  <user name="test2">
    <groups>
      <group name="ecss-biysk.local-domain-admin"/>
      <group name="ecss-user"/>
      <group name="ecss-admin"/>
    </groups>
  </user>
</in>

```

Ответ:

200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_del_user_to_group.xsd">
  <result>
    <ok user="test_user" group="ecss-admin"/>
    <ok user="test2" group="ecss-biysk.local-domain-admin"/>
    <ok user="test2" group="ecss-user"/>
    <ok user="test2" group="ecss-admin"/>
  </result>
</out>

```

[Hc_my_groups](#)

Команда позволяет вывести список групп доступа, разрешенных для текущего пользователя.

Аналог команды в CoCon:

`cocon/my-groups`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/my_groups`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- groupType -->
  <xs:complexType name="groupType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType/>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="groups">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="group" type="groupType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/my_groups

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_my_groups.xsd">
  <request/>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_my_groups.xsd">
  <groups>
    <group name="ecss-admin"/>
    <group name="ecss-biysk.local-domain-admin"/>
    <group name="ecss-biysk.local-domain-user"/>
    <group name="ecss-ct.office-domain-admin"/>
    <group name="ecss-ct.office-domain-user"/>
    <group name="ecss-user"/>
  </groups>
</out>
```

Нс_cocon_passwd

Команда позволяет сменить пароль для текущего пользователя CoCon.

Аналог команды в CoCon:

cocon/passwd

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cocon_passwd

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 200 – в случае ошибки, с описанием ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- errorType-->
  <xs:complexType name="errorType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
    <xs:attribute name="body" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="user">
          <xs:complexType>
            <xs:attribute name="name" type="xs:string" use="required"/>
            <xs:attribute name="password" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <!-- error -->
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/cocon_passwd

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_passwd.xsd">
  <user password="test_user" name="test123"/>
</in>

```

Ответ:

- в случае успеха: 201;
- в случае ошибки:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="cocon_passwd.xsd">
  <error name="testSSW" reason="No such object"/>
</out>
```

Нс_cocon_check_passwd

Команда позволяет проверить пароль для определенного пользователя CoCon.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cocon_check_passwd

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха. Если пароль неверный, возвращается 200 сообщение с *result correct="false"*;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- errorType-->
  <xs:complexType name="resultType">
    <xs:attribute name="correct" type="xs:boolean" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="user">
          <xs:complexType>
            <xs:attribute name="name" type="xs:string" use="required"/>
            <xs:attribute name="password" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/cocon_check_passwd

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_check_passwd.xsd">
  <user password="qwe123" name="test_user"/>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_check_passwd.xsd">
  <result correct="true"/>
</out>
```

[Hc_cocon_users_list](#)

Команда возвращает список пользователей подсистемы управления CoCon с указанием групп доступа, в которые каждый пользователь входит.

Аналог команды в CoCon:

cocon/list users

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cocon_users_list

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- groupType -->
  <xs:complexType name="groupType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- groupsType -->
  <xs:complexType name="groupsType">
    <xs:sequence>
      <xs:element name="group" type="groupType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- userType -->
  <xs:complexType name="userType">
    <xs:sequence>
      <xs:element name="groups" type="groupsType" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="role" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType/>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="users">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="user" type="userType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/cocon_users_list

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="cocon_users_list.xsd">
  <request/>
</in>
```

Ответ:

200


```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_users_list.xsd">
  <users>
    <user name="admin">
      <groups>
        <group name="ecss-admin"/>
        <group name="ecss-biysk.local-domain-admin"/>
        <group name="ecss-biysk.local-domain-user"/>
        <group name="ecss-ct.office-domain-admin"/>
        <group name="ecss-ct.office-domain-user"/>
        <group name="ecss-user"/>
      </groups>
    </user>
    <user name="support">
      <groups>
        <group name="ecss-admin"/>
        <group name="ecss-user"/>
      </groups>
    </user>
    <user name="test">
      <groups>
        <group name="ecss-biysk.local-domain-admin"/>
        <group name="ecss-biysk.local-domain-user"/>
      </groups>
    </user>
    <user name="test2">
      <groups>
        <group name="ecss-biysk.local-domain-user"/>
        <group name="ecss-ct.office-domain-admin"/>
        <group name="ecss-ct.office-domain-user"/>
      </groups>
    </user>
    <user name="test3">
      <groups/>
    </user>
    <user name="test_user">
      <groups>
        <group name="ecss-biysk.local-domain-admin"/>
        <group name="ecss-biysk.local-domain-user"/>
        <group name="ecss-user"/>
      </groups>
    </user>
    <user name="tst3">
      <groups/>
    </user>
    <user name="txtuser">
      <groups>
```

```
<group name="ecss-admin"/>
<group name="ecss-biysk.local-domain-admin"/>
<group name="ecss-biysk.local-domain-user"/>
<group name="ecss-ct.office-domain-admin"/>
<group name="ecss-ct.office-domain-user"/>
<group name="ecss-user"/>
</groups>
</user>
</users>
</out>
```

Ис_cocon_groups_list

Команда возвращает список групп доступа подсистемы CoCon с указанием пользователей, которые входят в данные группы.

Аналог команды в CoCon:

```
cocon/list groups
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cocon_groups_list

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- groupType -->
  <xs:complexType name="userType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- groupsType -->
  <xs:complexType name="usersType">
    <xs:sequence>
      <xs:element name="user" type="userType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- userType -->
  <xs:complexType name="groupType">
    <xs:sequence>
      <xs:element name="users" type="usersType" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType/>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="groups">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="group" type="groupType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/cocon_groups_list

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="cocon_groups_list.xsd">  
    <request />  
</in>
```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_groups_list.xsd">
  <groups>
    <group name="ecss-admin">
      <users>
        <user name="admin"/>
        <user name="support"/>
        <user name="test_user"/>
        <user name="txtuser"/>
      </users>
    </group>
    <group name="ecss-biysk.local-domain-admin">
      <users>
        <user name="admin"/>
        <user name="test2"/>
        <user name="txtuser"/>
      </users>
    </group>
    <group name="ecss-biysk.local-domain-user">
      <users>
        <user name="admin"/>
        <user name="test2"/>
        <user name="test_user"/>
        <user name="txtuser"/>
      </users>
    </group>
    <group name="ecss-ct.office-domain-admin">
      <users>
        <user name="admin"/>
        <user name="test_user"/>
      </users>
    </group>
    <group name="ecss-ct.office-domain-user">
      <users>
        <user name="admin"/>
        <user name="test_user"/>
      </users>
    </group>
    <group name="ecss-user">
      <users>
        <user name="admin"/>
        <user name="support"/>
        <user name="test2"/>
        <user name="test_user"/>
        <user name="txtuser"/>
      </users>
    </group>
  </groups>
</out>
```

```
<group name="{ECSS-ROOT}">
  <users/>
</group>
</groups>
</out>
```

Нс_cocon_roles_list

Команда возвращает список ролей подсистемы управления CoCon с указанием групп доступа, а также доменов, в которых данная роль видна.

Замечание: если роль входит во все домены, то в ответе присутствует ровно одна запись с доменом, имя которого равно "*".

Аналог команды в CoCon:

cocon/list roles

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cocon_roles_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- groupType -->
  <xs:complexType name="groupType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- domainType -->
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- groupsType -->
  <xs:complexType name="groupsType">
    <xs:sequence>
      <xs:element name="group" type="groupType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- userType -->
  <xs:complexType name="domainsType">
    <xs:sequence>
      <xs:element name="groups" type="domainType" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType/>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="roles">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="groups" type="groupsType" minOccurs="0"
maxOccurs="unbounded"/>
              <xs:element name="domains" type="domainsType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```
</xs:schema>
```

Пример

Запрос: http://192.168.1.22:9999/commands/cocon_roles_list

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="cocon_roles_list.xsd">
  <request/>
</in>
```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_roles_list.xsd">
  <roles>
    <role name="cc-admin">
      <groups/>
      <domains>
        <domain name="*" />
      </domains>
    </role>
    <role name="ecss-security">
      <groups/>
      <domains>
        <domain name="*" />
      </domains>
    </role>
    <role name="pbx-10X-admin">
      <groups/>
      <domains>
        <domain name="*" />
      </domains>
    </role>
  </roles>
</out>
```

[Hc_cocon_user_property_clean](#)

Команда позволяет удалять произвольные свойства CoCon пользователей.

Аналог команды в CoCon:

cocon/.properties/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cocon_user_property_clean

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 - в случае, если определенное свойство нельзя изменить;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!--requestType-->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!--propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример 1**Запрос:**

http://192.168.1.21:9999/commands/cocon_user_property_clean

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_user_property_clean.xsd">
  <request>
    <property key="test1"/>
    <property key="test2"/>
  </request>
</in>
```

Код ответа:

204

Пример 2

Запрос:

http://192.168.1.21:9999/commands/cocon_user_property_clean

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_user_property_clean.xsd">
  <request>
    <property key="shell_history_size"/>
    <property key="test2"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_user_property_clean.xsd">
  <error cmd="hc_cocon_user_property_clean" reason="restricted_property"
  body="shell_history_size"/>
</out>
```

[Hc_cocon_user_property_info](#)

Команда позволяет просматривать свойства CoCon пользователей.

Аналог команды в CoCon:

cocon/.properties/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/cocon_user_property_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!--requestType-->
  <xs:complexType name="requestType">
    <xs:all/>
    <xs:attribute name="property" type="xs:string" use="optional"/>
  </xs:complexType>
  <!--propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!--responseType-->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!--in-->
  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name = "out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="response" type="responseType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1

Запрос:

http://192.168.1.21:9999/commands/cocon_user_property_info

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_user_property_info.xsd">
  <request property="test1"/>
</in>

```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_user_property_info.xsd">
  <response>
    <property key="test1" value="value1"/>
  </response>
</out>
```

Пример 2

Запрос:

http://192.168.1.21:9999/commands/cocon_user_property_info

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_user_property_info.xsd">
  <request/>
</in>
```

Ответ:

200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_user_property_info.xsd">
  <response>
    <property key="prompt_prop" value="\l@n \p"/>
    <property key="shell_history" value="system/zmq/status system/media/resource/list
system/zmq/status domain/biysk.local/trace/show ls -la cocon/.properties/set
shell_history_size 300 ls -la cocon/.properties/ ls -la cocon ls -la cocon/.properties/set
cocon/.properties/info cocon/list users cocon/list domain/biysk.local/trace/show cluster/
mediator/md1/alarms/list all domain/biysk.local/trace/show domain/biysk.local/trace/list
domain/biysk.local/sorm/info sorm/info system/media/resource/list cluster/adaptier/sip1/
pcap-trace/stop all domain/biysk.local/restfs/list sounds records/2019_05_20/ domain/
biysk.local/trace/show --Te 7aef338b domain/biysk.local/trace/list cluster/adaptier/sip1/
pcap-trace/start all any domain/biysk.local/restfs/list sounds records/2019_05_20/ domain/
biysk.local/restfs/list sounds records/ cluster/core/core1/ai/info system-status cocon/my-
groups domain/biysk.local/trace/list domain/biysk.local/cfc/info domain/biysk.local/cfc/
clean domain/biysk.local/cfc/info restfs/list domain/biysk.local/trace/list cluster/
mediator/md1/alarms/list all domain/biysk.local/ss/dialer/list domain/biysk.local/ss/
dialer/remove 1 domain/biysk.local/ss/dialer/list system-status cocon/passwd test_user
domain/biysk.local/cdr/make_finalize_cdr bsk3 domain/biysk.local/tts/properties/set
sign_cdr_enabled true system-status cluster/mediator/md1/alarms/list all cluster/mediator/
md1/alarms/clear all cluster/mediator/md1/alarms/list all system-status domain/
biysk.local/ss/dialer/info 4 --show-options --active true domain/biysk.local/ss/dialer/
info 4 --show-options cluster/core/core1/dialer/db/info domain/biysk.local/ss/dialer/info
4 --show-options domain/biysk.local/ss/dialer/info 4 domain/biysk.local/ss/dialer/declare
test3 debt_tmpl_2 manual --group debt_rating_3 --calls_limit 2 --lines_limit 2 --
recall_timeout 20 domain/biysk.local/ss/dialer/list domain/biysk.local/ss/dialer/info 3 --
active true domain/biysk.local/trace/list system-status domain/biysk.local/ss/dialer/start
3 now domain/biysk.local/calls/list domain/biysk.local/ss/dialer/declare test2 debt_tmpl_2
manual --group debt_rating_3 --calls_limit 2 domain/biysk.local/ss/dialer/template/set
debt_tmpl_2 --time_interval 07 00-23 00 domain/biysk.local/ss/dialer/template/list domain/
biysk.local/ss/dialer/list system/media/resource/list all cluster/mediator/md1/alarms/
clear all system-status system/media/resource/list all system/calls/domains учше system/
media/resource/list all cocon/passwd test_user system-status node/core1@ecss1/log/config/
rule-off ecss_zmq_transport_bin node/core1@ecss1/log/config/rule-on ecss_zmq_transport_bin
node/core1@ecss1/log/config/show-rule system-status system/media/resource/list all system/
media/resource/list system-status cocon/list users domain/biysk.local/ss/dialer/list
domain/biysk.local/lists/declare debt_rating_3 default debt-3 240501 240502 416977 240464
240101 domain/biysk.local/ss/dialer/info 1 --active true domain/biysk.local/ss/dialer/
start 1 now domain/biysk.local/ss/dialer/start 1 domain/biysk.local/ss/dialer/info 1
domain/biysk.local/ss/dialer/list system-status domain/biysk.local/trace/show --short-
payload domain/biysk.local/trace/list domain/biysk.local/properties/restrictions/set
dialer\channels 2 domain/biysk.local/properties/restrictions/info cocon/list users domain/
biysk.local/tts/properties/info domain/biysk.local/tts/properties/info manual_cdr_columns
cocon/list users node/core1@ecss1/log/config/show-rule domain/biysk.local/ss/dialer/info 2
--active true domain/biysk.local/trace/list "/>
    <property key="shell_history_size" value="100"/>
    <property key="test1" value="value1"/>
    <property key="test2" value="value2"/>
    <property key="test3" value="value31 value32"/>
  </response>
</out>

```

Ис_cocon_user_property_set

Команда позволяет выставлять/изменять произвольные свойства CoCon пользователей.

Аналог команды в CoCon:

cocon/.properties/set**Метод HTTP-запроса:**

POST

Шаблон HTTP-запроса:http://host:port/commands/cocon_user_property_set**Код HTTP-ответа:**

- 201 – в случае успеха;
- 200 – в случае, если определенное свойство нельзя изменить;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!--requestType-->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!--propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!--in-->
  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример 1**Запрос:**http://192.168.1.21:9999/commands/cocon_user_property_set

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_user_property_set.xsd">
  <request>
    <property key="test1" value="value1"/>
    <property key="test2" value="value2"/>
    <property key="test3" value="value31 value32"/>
  </request>
</in>
```

Код ответа:

201

Пример 2

Запрос:

http://192.168.1.21:9999/commands/cocon_user_property_set

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="cocon_user_property_set.xsd">
  <request>
    <property key="shell_history_size" value="300"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0" encoding="UTF-8"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="cocon_user_property_set.xsd">
  <error cmd="hc_cocon_user_property_set" reason="restricted_property"
  body="shell_history_size"/>
</out>
```

hc_conference_list - Команда просмотра участников конференции/ конференц-комнаты

Команда предназначена для просмотра списка собранных конференций/конференц-комнат, а также их участников.

Аналог команды в CoCon:

domain/<DOMAIN>/conference/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/conference_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:complexType name="Participant">
    <xs:attribute name="participant number" use="required"/>
  </xs:complexType>

  <xs:complexType name="Participants">
    <xs:sequence>
      <xs:element name="participant number" type="Participant" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- Conference -->
  <xs:complexType name="Conference">
    <xs:sequence>
      <xs:element name="participants" type="Participants"/>
    </xs:sequence>
    <xs:attribute name="space" type="xs:string" use="required"/>
    <xs:attribute name="room" type="xs:string" use="required"/>
    <xs:attribute name="legs_count" type="xs:nonNegativeInteger" use="required"/>
    <xs:attribute name="calls_limit" type="xs:string" use="required"/>
    <xs:attribute name="destroy_mode" type="xs:string" use="required"/>
    <xs:attribute name="timestamp" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- Conferences-->
  <xs:complexType name="Conferences">
    <xs:sequence>
      <xs:element name="conference" type="Conference" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" minOccurs="1">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="addr" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>

```

```

    <xs:all>
      <xs:element name="response" minOccurs="1">
        <xs:complexType>
          <xs:all>
            <xs:element name="conferences" type="Conferences"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1:

Запрос:

http://192.168.1.21:9999/commands/conference_list

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_confere
nce_list.xsd">
  <request domain="biysk.local" addr="3854240101"/>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_confere
nce_list.xsd">
  <response>
    <conferences>
      <conference space="chat_room" room="*C71*123#" legs_count="3" calls_limit="infinity"
destroy_mode="destroy_by_no_more_calls" timestamp="11.03.2021 11:52:41">
        <participants>
          <participant number="3854240466"/>
          <participant number="3854240101"/>
          <participant number="3854416977"/>
        </participants>
      </conference>
    </conferences>
  </response>
</out>

```

Нс_core_ai - Команды управления акустическими сигналами

- [Нс_core_ai_clean](#)
- [Нс_core_ai_info](#)
- [Нс_core_ai_set](#)

Нс_core_ai_clean

Команда для очистки свойств автоответчика.

Аналог команд в CoSop:

domain/<DOMAIN>/core/ai/clean

cluster/core/<CORE>/ai/clean

Шаблон URL-запроса:

http://host:port/commands/core_ai_clean

Коды ответа:

- 200 - успех;
- 404 - ошибка.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="core_ai_clean">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean Core domain properties (auto-informers).
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/core_ai_clean

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_ai_cl
ean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="cc_agent_already_on"/>
        <option name="wait_time"/>
        <option name="number_not_found"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_ai_cl
ean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="cc_agent_already_on"/>
        <success option="number_not_found"/>
        <success option="wait_time"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_core_ai_info

Команда для получения информации об автоответчиках.

Аналог команды в CoCon:

domain/<DOMAIN>/core/ai/info

Шаблон URL-запроса:

http://host:port/commands/core_ai_info

Коды ответа:

- 200 - успех;
- 404 - ошибка.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="core_ai_info">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show Core domain properties (auto-informers).
      Copyright (c) 2015, Eltex. All rights reserved.
      Tone id format:
        tone://?f2=425&off_msec=4000&on_msec=1000&f1=0
        multicast://IP:PORT?codec=CODEC
        system://PATH/FILE.wav
        domain://PATH/FILE.wav
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <xs:complexType name="coreAIOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="beep" type="xs:string" use="optional" />
        <xs:attribute name="dialtone" type="xs:string" use="optional" />
        <xs:attribute name="alerting" type="xs:string" use="optional" />
        <xs:attribute name="busy" type="xs:string" use="optional" />
        <xs:attribute name="error" type="xs:string" use="optional" />
        <xs:attribute name="alarm_one_digit" type="xs:string" use="optional" />
        <xs:attribute name="alarm_multi_digits" type="xs:string" use="optional" />
        <xs:attribute name="wait_connection" type="xs:string" use="optional" />
        <xs:attribute name="wait_time" type="xs:string" use="optional" />
        <xs:attribute name="number_not_found" type="xs:string" use="optional" />
        <xs:attribute name="number_not_detected" type="xs:string" use="optional" />
        <xs:attribute name="last_incoming_number" type="xs:string" use="optional" />
        <xs:attribute name="my_number" type="xs:string" use="optional" />
        <xs:attribute name="ss_activated" type="xs:string" use="optional" />
        <xs:attribute name="autoredial_with_cb_activated" type="xs:string" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <xs:attribute name="ss_changed" type="xs:string" use="optional" />
  <xs:attribute name="ss_deactivated" type="xs:string" use="optional" />
  <xs:attribute name="ss_deactivated_all" type="xs:string" use="optional" />
  <xs:attribute name="ss_already_activated" type="xs:string" use="optional" />
  <xs:attribute name="ss_activate_error" type="xs:string" use="optional" />
  <xs:attribute name="ss_deactivate_error" type="xs:string" use="optional" />
  <xs:attribute name="ss_test_error" type="xs:string" use="optional" />
  <xs:attribute name="pin_incorrect" type="xs:string" use="optional" />
  <xs:attribute name="enter_pin" type="xs:string" use="optional" />
  <xs:attribute name="enter_number_sharp" type="xs:string" use="optional" />
  <xs:attribute name="enter_pin_number_sharp" type="xs:string" use="optional" />
  <xs:attribute name="to_turn_on" type="xs:string" use="optional" />
  <xs:attribute name="to_turn_off" type="xs:string" use="optional" />
</xs:schema>

```

```

<xs:attribute name="cc_auth_incorrect" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_already_on" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_number_occupied" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_login_ok" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_logged_out" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_busy" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_auxwork" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_auxwork_planned" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_make_available" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_conversations_completed" type="xs:string" use="optio
nal" />
<xs:attribute name="ms_ringback" type="xs:string" use="optional" />
<xs:attribute name="ms_stay_on_line" type="xs:string" use="optional" />
<xs:attribute name="ms_intercom_announce" type="xs:string" use="optional" />
<xs:attribute name="ms_music_on_hold" type="xs:string" use="optional" />
<xs:attribute name="ms_call_waiting" type="xs:string" use="optional" />
<xs:attribute name="ms_call_waiting_ringback" type="xs:string" use="optional" />
<xs:attribute name="ms_busy" type="xs:string" use="optional" />
<xs:attribute name="ms_one_hold_busy" type="xs:string" use="optional" />
<xs:attribute name="ms_two_hold_busy" type="xs:string" use="optional" />
<xs:attribute name="ms_one_hold_collect" type="xs:string" use="optional" />
<xs:attribute name="ms_two_hold_collect" type="xs:string" use="optional" />
<xs:attribute name="ms_intercom_end" type="xs:string" use="optional" />
<xs:attribute name="ms_intercom_tone" type="xs:string" use="optional" />
<xs:attribute name="zero" type="xs:string" use="optional" />
<xs:attribute name="one" type="xs:string" use="optional" />
<xs:attribute name="one_n" type="xs:string" use="optional" />
<xs:attribute name="one_et" type="xs:string" use="optional" />
<xs:attribute name="two" type="xs:string" use="optional" />
<xs:attribute name="two_et" type="xs:string" use="optional" />
<xs:attribute name="three" type="xs:string" use="optional" />
<xs:attribute name="four" type="xs:string" use="optional" />
<xs:attribute name="five" type="xs:string" use="optional" />
<xs:attribute name="six" type="xs:string" use="optional" />
<xs:attribute name="seven" type="xs:string" use="optional" />
<xs:attribute name="eight" type="xs:string" use="optional" />
<xs:attribute name="nine" type="xs:string" use="optional" />
<xs:attribute name="star" type="xs:string" use="optional" />
<xs:attribute name="star1" type="xs:string" use="optional" />
<xs:attribute name="sharp" type="xs:string" use="optional" />
<xs:attribute name="sharp1" type="xs:string" use="optional" />
<xs:attribute name="push_sharp" type="xs:string" use="optional" />
<xs:attribute name="push_star" type="xs:string" use="optional" />
<xs:attribute name="push_zero" type="xs:string" use="optional" />
<xs:attribute name="push_one" type="xs:string" use="optional" />
<xs:attribute name="push_two" type="xs:string" use="optional" />
<xs:attribute name="push_three" type="xs:string" use="optional" />
<xs:attribute name="push_four" type="xs:string" use="optional" />
<xs:attribute name="push_five" type="xs:string" use="optional" />
<xs:attribute name="push_six" type="xs:string" use="optional" />

```



```

<xs:attribute name="push_seven" type="xs:string" use="optional" />
<xs:attribute name="push_eight" type="xs:string" use="optional" />
<xs:attribute name="push_nine" type="xs:string" use="optional" />
<xs:attribute name="voicemail_listen_prompt" type="xs:string" use="optional" />
<xs:attribute name="voicemail_mailbox_empty" type="xs:string" use="optional" />
<xs:attribute name="voicemail_you_have" type="xs:string" use="optional" />
<xs:attribute name="voicemail_new_message" type="xs:string" use="optional" />
<xs:attribute name="voicemail_new_messages1" type="xs:string" use="optional" />
<xs:attribute name="voicemail_new_messages2" type="xs:string" use="optional" />
<xs:attribute name="voicemail_old_message" type="xs:string" use="optional" />
<xs:attribute name="voicemail_old_messages1" type="xs:string" use="optional" />
<xs:attribute name="voicemail_old_messages2" type="xs:string" use="optional" />
<xs:attribute name="voicemail_saved_message" type="xs:string" use="optional" />
<xs:attribute name="voicemail_saved_messages1" type="xs:string" use="optional" />
<xs:attribute name="voicemail_saved_messages2" type="xs:string" use="optional" />
<xs:attribute name="voicemail_listen_new_messages" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_listen_old_messages" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_listen_save_messages" type="xs:string" use="optiona
l" />
<xs:attribute name="voicemail_remove_new_messages" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_remove_old_messages" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_remove_save_messages" type="xs:string" use="optiona
l" />
<xs:attribute name="voicemail_confirm_remove_messages" type="xs:string" use="opti
onal" />
<xs:attribute name="voicemail_decline_remove_messages" type="xs:string" use="opti
onal" />
<xs:attribute name="voicemail_no_more_messages" type="xs:string" use="optional" /
>
<xs:attribute name="voicemail_play_file_delimiter" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_messages_successfully_removed" type="xs:string" use="
optional" />
<xs:attribute name="for_listen_next_message" type="xs:string" use="optional" />
<xs:attribute name="voicemail_play_back_message" type="xs:string" use="optional"
/>
<xs:attribute name="voicemail_play_forward_message" type="xs:string" use="optiona
l" />
<xs:attribute name="voicemail_replay_message" type="xs:string" use="optional" />
<xs:attribute name="voicemail_remove_current_message" type="xs:string" use="optio
nal" />
<xs:attribute name="voicemail_save_current_message" type="xs:string" use="optiona
l" />
<xs:attribute name="voicemail_listen_current_message" type="xs:string" use="optio
nal" />
<xs:attribute name="voicemail_message_saved" type="xs:string" use="optional" />

```

```

<xs:attribute name="voicemail_call_to_sender" type="xs:string" use="optional" />
<xs:attribute name="voicemail_pause_resume_message" type="xs:string" use="optiona
1" />
<xs:attribute name="voicemail_play_message_details" type="xs:string" use="optiona
1" />
<xs:attribute name="voicemail_ai_skip_message" type="xs:string" use="optional" />
<xs:attribute name="voicemail_help" type="xs:string" use="optional" />
<xs:attribute name="voicemail_go_back" type="xs:string" use="optional" />
<xs:attribute name="voicemail_exit" type="xs:string" use="optional" />
<xs:attribute name="voicemail_message_from_number" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_enter_password" type="xs:string" use="optional" />
<xs:attribute name="voicemail_enter_number_sharp" type="xs:string" use="optional"
/>
<xs:attribute name="voicemail_try_once_more" type="xs:string" use="optional" />
<xs:attribute name="voicemail_password_incorrect" type="xs:string" use="optional"
/>
<xs:attribute name="voicemail_message_successfully_removed" type="xs:string" use="
optional" />
<xs:attribute name="voicemail_ss_unavailable" type="xs:string" use="optional" />
<xs:attribute name="voicemail_number_absent" type="xs:string" use="optional" />
<xs:attribute name="voicemail_change_service_settings" type="xs:string" use="opti
onal" />
<xs:attribute name="voicemail_listen_messages" type="xs:string" use="optional" />
<xs:attribute name="voicemail_change_write_conditions" type="xs:string" use="opti
onal" />
<xs:attribute name="voicemail_change_common_settings" type="xs:string" use="optio
nal" />
<xs:attribute name="voicemail_mgm_intro" type="xs:string" use="optional" />
<xs:attribute name="voicemail_change_password" type="xs:string" use="optional" />
<xs:attribute name="voicemail_change_send_by_email" type="xs:string" use="optiona
1" />
<xs:attribute name="voicemail_change_play_details" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_enter_current_password" type="xs:string" use="optio
nal" />
<xs:attribute name="voicemail_enter_new_password" type="xs:string" use="optional"
/>
<xs:attribute name="voicemail_reenter_new_password" type="xs:string" use="optiona
1" />
<xs:attribute name="voicemail_passwords_different" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_password_successfully_changed" type="xs:string" use="
optional" />
<xs:attribute name="voicemail_property_changed" type="xs:string" use="optional" /
>
<xs:attribute name="voicemail_change_write_by_no_reply" type="xs:string" use="opt
ional" />
<xs:attribute name="voicemail_change_write_by_busy" type="xs:string" use="optiona
1" />

```

```

        <xs:attribute name="voicemail_change_write_by_out_of_service" type="xs:string"
use="optional" />
        <xs:attribute name="voicemail_change_write_by_unconditional" type="xs:string" use="
optional" />
        <xs:attribute name="voicemail_remove_user_intro" type="xs:string" use="optional"
/>
        <xs:attribute name="voicemail_write_user_intro" type="xs:string" use="optional" /
>
        <xs:attribute name="voicemail_say_message_after_beep" type="xs:string" use="optio
nal" />
        <xs:attribute name="voicemail_change_write_by_dnd" type="xs:string" use="optiona
l" />

        <xs:attribute name="monday" type="xs:string" use="optional" />
        <xs:attribute name="tuesday" type="xs:string" use="optional" />
        <xs:attribute name="wednesday" type="xs:string" use="optional" />
        <xs:attribute name="wednesday2" type="xs:string" use="optional" />
        <xs:attribute name="thursday" type="xs:string" use="optional" />
        <xs:attribute name="friday" type="xs:string" use="optional" />
        <xs:attribute name="friday2" type="xs:string" use="optional" />
        <xs:attribute name="saturday" type="xs:string" use="optional" />
        <xs:attribute name="saturday2" type="xs:string" use="optional" />
        <xs:attribute name="sunday" type="xs:string" use="optional" />
        <xs:attribute name="alarm_on" type="xs:string" use="optional" />
        <xs:attribute name="alarm_off" type="xs:string" use="optional" />
        <xs:attribute name="alarm_today_off" type="xs:string" use="optional" />
        <xs:attribute name="nearest1" type="xs:string" use="optional" />
        <xs:attribute name="nearest2" type="xs:string" use="optional" />
        <xs:attribute name="nearest3" type="xs:string" use="optional" />
        <xs:attribute name="nearest4" type="xs:string" use="optional" />
        <xs:attribute name="every1" type="xs:string" use="optional" />
        <xs:attribute name="every2" type="xs:string" use="optional" />
        <xs:attribute name="every3" type="xs:string" use="optional" />
        <xs:attribute name="every4" type="xs:string" use="optional" />
        <xs:attribute name="call_forwarding" type="xs:string" use="optional" />
        <xs:attribute name="from" type="xs:string" use="optional" />
        <xs:attribute name="untill" type="xs:string" use="optional" />
        <xs:attribute name="activated" type="xs:string" use="optional" />
        <xs:attribute name="second1" type="xs:string" use="optional" />
        <xs:attribute name="second2" type="xs:string" use="optional" />
        <xs:attribute name="second3" type="xs:string" use="optional" />
        <xs:attribute name="number_is_hidden" type="xs:string" use="optional" />
        <xs:attribute name="autoredial_with_cb_inactive" type="xs:string" use="optional"
/>
        <xs:attribute name="autoredial_with_cb_deactivated" type="xs:string" use="optiona
l" />

        <xs:attribute name="ms_ringback_second" type="xs:string" use="optional" />
        <xs:attribute name="ms_release_busy" type="xs:string" use="optional" />
        <xs:attribute name="ms_preemption_busy" type="xs:string" use="optional" />
        <xs:attribute name="call_recording_notification" type="xs:string" use="optional"
/>

```

```

    <xs:attribute name="call_recording_announcement" type="xs:string" use="optional"
  />
    <xs:attribute name="cc_supervising_mode_selected" type="xs:string" use="optional"
  />
    <xs:attribute name="cc_supervising_mode_observing" type="xs:string" use="optional
" />
    <xs:attribute name="cc_supervising_mode_consult" type="xs:string" use="optional"
  />
    <xs:attribute name="cc_supervising_mode_conference" type="xs:string" use="optiona
1" />
    <xs:attribute name="teleconference_greeting" type="xs:string" use="optional" />
    <xs:attribute name="teleconference_farewell" type="xs:string" use="optional" />
    <xs:attribute name="teleconference_join" type="xs:string" use="optional" />
    <xs:attribute name="teleconference_leave" type="xs:string" use="optional" />
    <xs:attribute name="conference_greeting" type="xs:string" use="optional" />
    <xs:attribute name="conference_farewell" type="xs:string" use="optional" />
    <xs:attribute name="conference_full" type="xs:string" use="optional" />
    <xs:attribute name="conference_restriction" type="xs:string" use="optional" />
    <xs:attribute name="conference_already_exists" type="xs:string" use="optional" />
    <xs:attribute name="conference_not_exists" type="xs:string" use="optional" />
    <xs:attribute name="conference_join" type="xs:string" use="optional" />
    <xs:attribute name="conference_leave" type="xs:string" use="optional" />
    <xs:attribute name="voicemail_message_from_hidden_number" type="xs:string" use="o
ptional" />
    <xs:attribute name="zone_page_ready_tone" type="xs:string" use="optional" />
    <xs:attribute name="exact_time" type="xs:string" use="optional" />
    <xs:attribute name="mailbox_empty" type="xs:string" use="optional" />
    <xs:attribute name="ringback" type="xs:string" use="optional" />
    <xs:attribute name="subscriber_do_not_disturb" type="xs:string" use="optional" />
  </xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoInType" minOccurs="
1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>

```

```

        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос значений всех доступных свойств:

http://192.168.1.21:9999/commands/core_ai_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_ai_in
fo.xsd">
  <request>
    <domain_options domain="biysk.local" />
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_ai_in
fo.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="coreAIOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="coreAIOptionsType" zone_page_ready_tone="tone://?
f2=525&off_msec=500&on_msec=500&f1=525&duration=1000" zero="system://sounds/numbers/0.wav"
wednesday2="system://sounds/dow/ai_wednesday2.wav" wednesday="system://sounds/dow/wednesday.wav"
wait_time="system://sounds/ai_wait_time.wav" wait_connection="system://sounds/
ai_wait_connection.wav" voicemail_you_have="system://sounds/voicemail/ai_you_have.wav"
voicemail_write_user_intro="system://sounds/voicemail/ai_voicemail_write_user_intro.wav"
voicemail_try_once_more="system://sounds/voicemail/ai_try_once_more.wav" voicemail_ss_unavailable="
system://sounds/voicemail/ai_ss_unavailable.wav" voicemail_say_message_after_beep="system://

```

```

sounds/voicemail/ai_say_message_after_beep.wav" voicemail_saved_messages2="system://sounds/
voicemail/ai_saved_messages_2.wav" voicemail_saved_messages1="system://sounds/voicemail/
ai_saved_messages_1.wav" voicemail_saved_message="system://sounds/voicemail/ai_saved_message.wav"
voicemail_save_current_message="system://sounds/voicemail/ai_save_current_message.wav"
voicemail_replay_message="system://sounds/voicemail/ai_replay_message.wav"
voicemail_remove_user_intro="system://sounds/voicemail/ai_voicemail_remove_user_intro.wav"
voicemail_remove_save_messages="system://sounds/voicemail/ai_remove_save_messages.wav"
voicemail_remove_old_messages="system://sounds/voicemail/ai_remove_old_messages.wav"
voicemail_remove_new_messages="system://sounds/voicemail/ai_remove_new_messages.wav"
voicemail_remove_current_message="system://sounds/voicemail/ai_remove_current_message.wav"
voicemail_reenter_new_password="system://sounds/voicemail/ai_reenter_new_password.wav"
voicemail_reach_max_duration="system://sounds/voicemail/ai_voicemail_reach_max_duration.wav"
voicemail_property_changed="system://sounds/ai_changed.wav" voicemail_play_message_details="system://sounds/voicemail/ai_play_message_details.wav" voicemail_play_forward_message="system://sounds/voicemail/ai_play_forward_message.wav" voicemail_play_file_delimiter="tone://?f2=525&off_msec=500&on_msec=500&f1=525&duration=1000" voicemail_play_back_message="system://sounds/voicemail/ai_play_back_message.wav" voicemail_pause_resume_message="system://sounds/voicemail/ai_pause_resume_message.wav" voicemail_passwords_different="system://sounds/voicemail/ai_passwords_different.wav" voicemail_password_successfully_changed="system://sounds/voicemail/ai_password_successfully_changed.wav" voicemail_password_incorrect="system://sounds/voicemail/ai_voicemail_password_incorrect.wav" voicemail_old_messages2="system://sounds/voicemail/ai_old_messages_2.wav" voicemail_old_messages1="system://sounds/voicemail/ai_old_messages_1.wav" voicemail_old_message="system://sounds/voicemail/ai_old_message.wav" voicemail_number_absent="system://sounds/ai_number_absent.wav" voicemail_no_more_messages="system://sounds/voicemail/ai_no_more_messages.wav" voicemail_new_messages2="system://sounds/voicemail/ai_new_messages_2.wav" voicemail_new_messages1="system://sounds/voicemail/ai_new_messages_1.wav" voicemail_new_message="system://sounds/voicemail/ai_new_message.wav" voicemail_mgm_intro="system://sounds/voicemail/ai_voicemail_mgm_intro.wav" voicemail_messages_successfully_removed="system://sounds/voicemail/ai_messages_successfully_removed.wav" voicemail_message_successfully_removed="system://sounds/voicemail/ai_message_successfully_removed.wav" voicemail_message_saved="system://sounds/voicemail/ai_message_saved.wav" voicemail_message_from_number="system://sounds/voicemail/ai_message_from_number.wav" voicemail_message_from_hidden_number="system://sounds/voicemail/ai_message_from_hidden_number.wav" voicemail_mailbox_empty="system://sounds/voicemail/ai_voicemail_mailbox_empty.wav" voicemail_listen_save_messages="system://sounds/voicemail/ai_listen_save_messages.wav" voicemail_listen_prompt="system://sounds/voicemail/ai_listen_prompt.wav" voicemail_listen_old_messages="system://sounds/voicemail/ai_listen_old_messages.wav" voicemail_listen_new_messages="system://sounds/voicemail/ai_listen_new_messages.wav" voicemail_listen_messages="system://sounds/voicemail/ai_listen_messages.wav" voicemail_listen_current_message="system://sounds/voicemail/ai_listen_current_file.wav" voicemail_help="system://sounds/ai_for_help_at_any_time.wav" voicemail_go_back="system://sounds/ai_to_go_back_at_any_time.wav" voicemail_exit="system://sounds/voicemail/ai_exit.wav" voicemail_enter_password="system://sounds/ai_enter_password.wav" voicemail_enter_number_sharp="system://sounds/voicemail/ai_enter_number_sharp.wav" voicemail_enter_new_password="system://sounds/voicemail/ai_enter_new_password.wav" voicemail_enter_current_password="system://sounds/voicemail/ai_enter_current_password.wav" voicemail_decline_remove_messages="system://sounds/voicemail/ai_decline_remove_messages.wav" voicemail_confirm_remove_messages="system://sounds/voicemail/ai_confirm_remove_messages.wav" voicemail_change_write_conditions="system://sounds/voicemail/ai_voicemail_change_write_conditions.wav" voicemail_change_write_by_unconditional="system://sounds/voicemail/ai_voicemail_change_write_by_unconditional.wav" voicemail_change_write_by_out_of_service="system://sounds/voicemail/

```



```

ai_voicemail_change_write_by_out_of_service.wav" voicemail_change_write_by_no_reply="system://
sounds/voicemail/ai_voicemail_change_write_by_no_reply.wav" voicemail_change_write_by_dnd="system
://sounds/voicemail/ai_voicemail_change_write_by_dnd.wav" voicemail_change_write_by_busy="system:
//sounds/voicemail/ai_voicemail_change_write_by_busy.wav" voicemail_change_service_settings="syst
em://sounds/voicemail/ai_change_service_settings.wav" voicemail_change_send_by_email="system://
sounds/voicemail/ai_voicemail_change_send_by_email.wav" voicemail_change_play_details="system://
sounds/voicemail/ai_voicemail_change_play_details.wav" voicemail_change_password="system://
sounds/voicemail/ai_voicemail_change_password.wav" voicemail_change_common_settings="system://
sounds/voicemail/ai_voicemail_change_common_settings.wav" voicemail_call_to_sender="system://
sounds/voicemail/ai_call_to_sender.wav" voicemail_ai_skip_message="system://sounds/voicemail/
ai_skip_message.wav" untill="system://sounds/ai_until.wav" two_et="system://sounds/et/2.wav" two="
system://sounds/numbers/2.wav" tuesday="system://sounds/dow/tuesday.wav" to_turn_on="system://
sounds/ai_to_turn_on.wav" to_turn_off="system://sounds/ai_to_turn_off.wav" thursday="system://
sounds/dow/thursday.wav" three="system://sounds/numbers/3.wav" teleconference_leave="tone://?
f1=390&f2=0&on_msec=200&off_msec=100&duration=300" teleconference_join="tone://?
f1=490&f2=0&on_msec=200&off_msec=100&duration=300" teleconference_greeting="system://sounds/
ai_conference_prompt.wav" teleconference_farewell="system://sounds/ai_conference_destroyed.wav"
sunday="system://sounds/dow/sunday.wav" subscriber_do_not_disturb="system://sounds/
ai_subscriber_do_not_disturb.wav" star1="system://sounds/numbers/*1.wav" star="system://sounds/
numbers/*.wav" ss_test_error="system://sounds/ai_error.wav" ss_deactivated_all="system://sounds/
ai_deactivated_all.wav" ss_deactivated="system://sounds/ai_deactivated.wav" ss_deactivate_error="
system://sounds/ai_error.wav" ss_changed="system://sounds/ai_changed.wav" ss_already_activated="s
ystem://sounds/ai_already_activated.wav" ss_activated="system://sounds/ai_activated.wav"
ss_activate_error="system://sounds/ai_error.wav" six="system://sounds/numbers/6.wav" sharp1="syst
em://sounds/numbers/#1.wav" sharp="system://sounds/numbers/#.wav" seven="system://sounds/numbers/
7.wav" second3="system://sounds/et/second3.wav" second2="system://sounds/et/second2.wav" second1="
system://sounds/et/second1.wav" saturday2="system://sounds/dow/ai_saturday2.wav" saturday="system
://sounds/dow/saturday.wav" ringback="tone://?f2=425&off_msec=4000&on_msec=1000&f1=0&duration=-1"
push_zero="system://sounds/push_numbers/ai_push_0.wav" push_two="system://sounds/push_numbers/
ai_push_2.wav" push_three="system://sounds/push_numbers/ai_push_3.wav" push_star="system://
sounds/push_numbers/ai_push_*.wav" push_six="system://sounds/push_numbers/ai_push_6.wav"
push_sharp="system://sounds/push_numbers/ai_push_#.wav" push_seven="system://sounds/push_numbers/
ai_push_7.wav" push_one="system://sounds/push_numbers/ai_push_1.wav" push_nine="system://sounds/
push_numbers/ai_push_9.wav" push_four="system://sounds/push_numbers/ai_push_4.wav" push_five="sys
tem://sounds/push_numbers/ai_push_5.wav" push_eight="system://sounds/push_numbers/ai_push_8.wav"
pin_incorrect="system://sounds/ai_pin_incorrect.wav" one_n="system://sounds/numbers/1n.wav"
one_et="system://sounds/et/1.wav" one="system://sounds/numbers/1.wav" number_not_found="system://
sounds/ai_number_not_found.wav" number_not_detected="system://sounds/ai_number_not_detected.wav"
number_is_hidden="system://sounds/ai_number_is_hidden.wav" nine="system://sounds/numbers/9.wav"
nearest4="system://sounds/ai_nearest4.wav" nearest3="system://sounds/ai_nearest3.wav" nearest2="s
ystem://sounds/ai_nearest2.wav" nearest1="system://sounds/ai_nearest1.wav" my_number="system://
sounds/ai_my_number.wav" ms_two_hold_collect="tone://?
f2=425&off_msec=0&on_msec=1000&f1=0&duration=30000" ms_two_hold_busy="tone://?
f2=425&off_msec=400&on_msec=400&f1=0&duration=10000" ms_stay_on_line="system://sounds/
ai_stay_on_line.wav" ms_ringback_second="system://sounds/ai_ringback_second.wav" ms_ringback="ton
e://?f2=425&off_msec=4000&on_msec=1000&f1=0&duration=-1" ms_release_busy="tone://?
f2=425&off_msec=400&on_msec=400&f1=0&duration=2600" ms_preemption_busy="tone://?
f2=425&off_msec=400&on_msec=400&f1=0&duration=10000" ms_one_hold_collect="tone://?
f2=425&off_msec=0&on_msec=1000&f1=0&duration=30000" ms_one_hold_busy="tone://?
f2=425&off_msec=400&on_msec=400&f1=0&duration=10000" ms_music_on_hold="system://sounds/
ai_remote_hold.wav" ms_intercom_end="system://sounds/ai_intercom_end.wav" ms_intercom_announce="s

```

```

system://sounds/ai_intercom_announce.wav" ms_call_waiting_ringback="tone://?
f2=425&off_msec=4000&on_msec=1000&f1=0&duration=-1" ms_call_waiting="tone://?
f2=425&off_msec=5000&on_msec=200&f1=0&duration=-1" ms_busy="tone://?
f2=425&off_msec=400&on_msec=400&f1=0&duration=10000" monday="system://sounds/dow/monday.wav"
mailbox_empty="system://sounds/ai_mailbox_empty.wav" last_incoming_number="system://sounds/
ai_last_incoming_number.wav" from="system://sounds/ai_from.wav" friday2="system://sounds/dow/
ai_friday2.wav" friday="system://sounds/dow/friday.wav" four="system://sounds/numbers/4.wav"
for_listen_next_message="system://sounds/voicemail/ai_for_listen_next_message.wav" five="system:/
/sounds/numbers/5.wav" exact_time="system://sounds/ai_exact_time.wav" every4="system://sounds/
ai_every4.wav" every3="system://sounds/ai_every3.wav" every2="system://sounds/ai_every2.wav"
every1="system://sounds/ai_every1.wav" error="system://sounds/ai_error.wav"
enter_pin_number_sharp="system://sounds/ai_enter_pin_number_sharp.wav" enter_pin="system://
sounds/ai_enter_pin.wav" enter_number_sharp="system://sounds/ai_enter_number_sharp.wav" eight="sy
stem://sounds/numbers/8.wav" dialtone="tone://?f2=425&off_msec=0&on_msec=1000&f1=0"
conference_restriction="system://sounds/reached_conference_restriction.wav" conference_not_exists="
system://sounds/conference_not_exists.wav" conference_leave="tone://?
f1=390&f2=0&on_msec=200&off_msec=100&duration=300" conference_join="tone://?
f1=490&f2=0&on_msec=200&off_msec=100&duration=300" conference_greeting="system://sounds/
ai_conference_prompt.wav" conference_full="system://sounds/ai_conference_full.wav"
conference_farewell="system://sounds/ai_conference_destroyed.wav" conference_already_exists="syst
em://sounds/conference_already_exists.wav" cc_supervising_mode_selected="system://sounds/
ai_supervising_mode_selected.wav" cc_supervising_mode_observing="system://sounds/
ai_supervising_mode_observing.wav" cc_supervising_mode_consult="system://sounds/
ai_supervising_mode_consult.wav" cc_supervising_mode_conference="system://sounds/
ai_supervising_mode_conference.wav" cc_auth_incorrect="system://sounds/ai_auth_incorrect.wav"
cc_agent_number_occupied="system://sounds/ai_agent_number_occupied.wav" cc_agent_make_available="
system://sounds/ai_activated.wav" cc_agent_login_ok="system://sounds/ai_agent_login_ok.wav"
cc_agent_logged_out="system://sounds/ai_agent_logged_out.wav" cc_agent_conversations_completed="s
ystem://sounds/ai_activated.wav" cc_agent_busy="system://sounds/ai_agent_busy.wav"
cc_agent_auxwork_planned="system://sounds/ai_activated.wav" cc_agent_auxwork="system://sounds/
ai_agent_auxwork.wav" cc_agent_already_on="system://sounds/ai_agent_already_on.wav"
call_recording_notification="system://sounds/ai_call_recording_notification.wav"
call_recording_announcement="system://sounds/ai_call_recording_announcement.wav" call_forwarding="
system://sounds/ai_call_forwarding.wav" busy="tone://?
f2=425&off_msec=400&on_msec=400&f1=0&duration=300000" beep="tone://?
f1=525&f2=525&on_msec=500&off_msec=500&duration=1000" autoredial_with_cb_inactive="system://
sounds/ai_unavailable.wav" autoredial_with_cb_deactivated="system://sounds/ai_deactivated.wav"
autoredial_with_cb_activated="system://sounds/ai_activated.wav" alerting="tone://?
f2=425&off_msec=4000&on_msec=1000&f1=0" alarm_today_off="system://sounds/ai_alarm_off_today.wav"
alarm_one_digit="system://sounds/ai_alarm_one_digit.wav" alarm_on="system://sounds/
ai_alarm_on.wav" alarm_off="system://sounds/ai_alarm_off.wav" alarm_multi_digits="system://
sounds/ai_alarm_multi_digits.wav" activated="system://sounds/activated.wav"/>
    </domain>
  </domain_options>
</result>
</out>

```


Нс_core_ai_set

Команда для изменения свойств автоответчика.

Аналог команды в CoCon:

`domain/<DOMAIN>/core/ai/set`

Шаблон URL-запроса:

http://host:port/commands/core_ai_set

Коды ответа:

- 200 - успех;
- 404 - ошибка.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="core_ai_set">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Change Core domain properties (auto-informers).
      Copyright (c) 2015, Eltex. All rights reserved.
      Tone id format:
        tone://?f2=425&off_msec=4000&on_msec=1000&f1=0
        multicast://IP:PORT?codec=CODEC
        system://PATH/FILE.wav
        domain://PATH/FILE.wav
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <xs:complexType name="coreAIOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="beep" type="xs:string" use="optional" />
        <xs:attribute name="dialtone" type="xs:string" use="optional" />
        <xs:attribute name="alerting" type="xs:string" use="optional" />
        <xs:attribute name="busy" type="xs:string" use="optional" />
        <xs:attribute name="error" type="xs:string" use="optional" />
        <xs:attribute name="alarm_one_digit" type="xs:string" use="optional" />
        <xs:attribute name="alarm_multi_digits" type="xs:string" use="optional" />
        <xs:attribute name="wait_connection" type="xs:string" use="optional" />
        <xs:attribute name="wait_time" type="xs:string" use="optional" />
        <xs:attribute name="number_not_found" type="xs:string" use="optional" />
        <xs:attribute name="number_not_detected" type="xs:string" use="optional" />
        <xs:attribute name="last_incoming_number" type="xs:string" use="optional" />
        <xs:attribute name="my_number" type="xs:string" use="optional" />
        <xs:attribute name="autoredial_with_cb_activated" type="xs:string" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <xs:attribute name="ss_changed" type="xs:string" use="optional" />
  <xs:attribute name="ss_activated" type="xs:string" use="optional" />
  <xs:attribute name="ss_deactivated" type="xs:string" use="optional" />
  <xs:attribute name="ss_deactivated_all" type="xs:string" use="optional" />
  <xs:attribute name="ss_already_activated" type="xs:string" use="optional" />
  <xs:attribute name="ss_activate_error" type="xs:string" use="optional" />
  <xs:attribute name="ss_deactivate_error" type="xs:string" use="optional" />
  <xs:attribute name="ss_test_error" type="xs:string" use="optional" />
  <xs:attribute name="pin_incorrect" type="xs:string" use="optional" />
  <xs:attribute name="enter_pin" type="xs:string" use="optional" />
  <xs:attribute name="enter_number_sharp" type="xs:string" use="optional" />
  <xs:attribute name="enter_pin_number_sharp" type="xs:string" use="optional" />
  <xs:attribute name="to_turn_on" type="xs:string" use="optional" />
  <xs:attribute name="to_turn_off" type="xs:string" use="optional" />
</xs:schema>

```

```

<xs:attribute name="cc_auth_incorrect" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_already_on" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_number_occupied" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_login_ok" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_logged_out" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_busy" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_auxwork" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_auxwork_planned" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_make_available" type="xs:string" use="optional" />
<xs:attribute name="cc_agent_conversations_completed" type="xs:string" use="optio
nal" />
<xs:attribute name="ms_ringback" type="xs:string" use="optional" />
<xs:attribute name="ms_music_on_hold" type="xs:string" use="optional" />
<xs:attribute name="ms_stay_on_line" type="xs:string" use="optional" />
<xs:attribute name="ms_intercom_announce" type="xs:string" use="optional" />
<xs:attribute name="ms_call_waiting" type="xs:string" use="optional" />
<xs:attribute name="ms_call_waiting_ringback" type="xs:string" use="optional" />
<xs:attribute name="ms_busy" type="xs:string" use="optional" />
<xs:attribute name="ms_one_hold_busy" type="xs:string" use="optional" />
<xs:attribute name="ms_two_hold_busy" type="xs:string" use="optional" />
<xs:attribute name="ms_one_hold_collect" type="xs:string" use="optional" />
<xs:attribute name="ms_two_hold_collect" type="xs:string" use="optional" />
<xs:attribute name="ms_intercom_end" type="xs:string" use="optional" />
<xs:attribute name="ms_intercom_tone" type="xs:string" use="optional" />
<xs:attribute name="zero" type="xs:string" use="optional" />
<xs:attribute name="one" type="xs:string" use="optional" />
<xs:attribute name="one_n" type="xs:string" use="optional" />
<xs:attribute name="one_et" type="xs:string" use="optional" />
<xs:attribute name="two" type="xs:string" use="optional" />
<xs:attribute name="two_et" type="xs:string" use="optional" />
<xs:attribute name="three" type="xs:string" use="optional" />
<xs:attribute name="four" type="xs:string" use="optional" />
<xs:attribute name="five" type="xs:string" use="optional" />
<xs:attribute name="six" type="xs:string" use="optional" />
<xs:attribute name="seven" type="xs:string" use="optional" />
<xs:attribute name="eight" type="xs:string" use="optional" />
<xs:attribute name="nine" type="xs:string" use="optional" />
<xs:attribute name="star" type="xs:string" use="optional" />
<xs:attribute name="star1" type="xs:string" use="optional" />
<xs:attribute name="sharp" type="xs:string" use="optional" />
<xs:attribute name="sharp1" type="xs:string" use="optional" />
<xs:attribute name="push_sharp" type="xs:string" use="optional" />
<xs:attribute name="push_star" type="xs:string" use="optional" />
<xs:attribute name="push_zero" type="xs:string" use="optional" />
<xs:attribute name="push_one" type="xs:string" use="optional" />
<xs:attribute name="push_two" type="xs:string" use="optional" />
<xs:attribute name="push_three" type="xs:string" use="optional" />
<xs:attribute name="push_four" type="xs:string" use="optional" />
<xs:attribute name="push_five" type="xs:string" use="optional" />
<xs:attribute name="push_six" type="xs:string" use="optional" />

```

```

<xs:attribute name="push_seven" type="xs:string" use="optional" />
<xs:attribute name="push_eight" type="xs:string" use="optional" />
<xs:attribute name="push_nine" type="xs:string" use="optional" />
<xs:attribute name="voicemail_listen_prompt" type="xs:string" use="optional" />
<xs:attribute name="voicemail_mailbox_empty" type="xs:string" use="optional" />
<xs:attribute name="voicemail_you_have" type="xs:string" use="optional" />
<xs:attribute name="voicemail_new_message" type="xs:string" use="optional" />
<xs:attribute name="voicemail_new_messages1" type="xs:string" use="optional" />
<xs:attribute name="voicemail_new_messages2" type="xs:string" use="optional" />
<xs:attribute name="voicemail_old_message" type="xs:string" use="optional" />
<xs:attribute name="voicemail_old_messages1" type="xs:string" use="optional" />
<xs:attribute name="voicemail_old_messages2" type="xs:string" use="optional" />
<xs:attribute name="voicemail_saved_message" type="xs:string" use="optional" />
<xs:attribute name="voicemail_saved_messages1" type="xs:string" use="optional" />
<xs:attribute name="voicemail_saved_messages2" type="xs:string" use="optional" />
<xs:attribute name="voicemail_listen_new_messages" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_listen_old_messages" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_listen_save_messages" type="xs:string" use="optiona
l" />
<xs:attribute name="voicemail_remove_new_messages" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_remove_old_messages" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_remove_save_messages" type="xs:string" use="optiona
l" />
<xs:attribute name="voicemail_confirm_remove_messages" type="xs:string" use="opti
onal" />
<xs:attribute name="voicemail_decline_remove_messages" type="xs:string" use="opti
onal" />
<xs:attribute name="voicemail_no_more_messages" type="xs:string" use="optional" /
>
<xs:attribute name="voicemail_play_file_delimiter" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_messages_successfully_removed" type="xs:string" use="
optional" />
<xs:attribute name="for_listen_next_message" type="xs:string" use="optional" />
<xs:attribute name="voicemail_play_back_message" type="xs:string" use="optional"
/>
<xs:attribute name="voicemail_play_forward_message" type="xs:string" use="optiona
l" />
<xs:attribute name="voicemail_replay_message" type="xs:string" use="optional" />
<xs:attribute name="voicemail_remove_current_message" type="xs:string" use="optio
nal" />
<xs:attribute name="voicemail_save_current_message" type="xs:string" use="optiona
l" />
<xs:attribute name="voicemail_listen_current_message" type="xs:string" use="optio
nal" />
<xs:attribute name="voicemail_message_saved" type="xs:string" use="optional" />

```

```

<xs:attribute name="voicemail_call_to_sender" type="xs:string" use="optional" />
<xs:attribute name="voicemail_pause_resume_message" type="xs:string" use="optiona
1" />
<xs:attribute name="voicemail_play_message_details" type="xs:string" use="optiona
1" />
<xs:attribute name="voicemail_ai_skip_message" type="xs:string" use="optional" />
<xs:attribute name="voicemail_help" type="xs:string" use="optional" />
<xs:attribute name="voicemail_go_back" type="xs:string" use="optional" />
<xs:attribute name="voicemail_exit" type="xs:string" use="optional" />
<xs:attribute name="voicemail_message_from_number" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_enter_password" type="xs:string" use="optional" />
<xs:attribute name="voicemail_enter_number_sharp" type="xs:string" use="optional"
/>
<xs:attribute name="voicemail_try_once_more" type="xs:string" use="optional" />
<xs:attribute name="voicemail_password_incorrect" type="xs:string" use="optional"
/>
<xs:attribute name="voicemail_message_successfully_removed" type="xs:string" use="
optional" />
<xs:attribute name="voicemail_ss_unavailable" type="xs:string" use="optional" />
<xs:attribute name="voicemail_number_absent" type="xs:string" use="optional" />
<xs:attribute name="voicemail_change_service_settings" type="xs:string" use="opti
onal" />
<xs:attribute name="voicemail_listen_messages" type="xs:string" use="optional" />
<xs:attribute name="voicemail_change_write_conditions" type="xs:string" use="opti
onal" />
<xs:attribute name="voicemail_change_common_settings" type="xs:string" use="optio
nal" />
<xs:attribute name="voicemail_mgm_intro" type="xs:string" use="optional" />
<xs:attribute name="voicemail_change_password" type="xs:string" use="optional" />
<xs:attribute name="voicemail_change_send_by_email" type="xs:string" use="optiona
1" />
<xs:attribute name="voicemail_change_play_details" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_enter_current_password" type="xs:string" use="optio
nal" />
<xs:attribute name="voicemail_enter_new_password" type="xs:string" use="optional"
/>
<xs:attribute name="voicemail_reenter_new_password" type="xs:string" use="optiona
1" />
<xs:attribute name="voicemail_passwords_different" type="xs:string" use="optional
" />
<xs:attribute name="voicemail_password_successfully_changed" type="xs:string" use="
optional" />
<xs:attribute name="voicemail_property_changed" type="xs:string" use="optional" /
>
<xs:attribute name="voicemail_change_write_by_no_reply" type="xs:string" use="opt
ional" />
<xs:attribute name="voicemail_change_write_by_busy" type="xs:string" use="optiona
1" />

```

```

        <xs:attribute name="voicemail_change_write_by_out_of_service" type="xs:string"
use="optional" />
        <xs:attribute name="voicemail_change_write_by_unconditional" type="xs:string" use="
optional" />
        <xs:attribute name="voicemail_change_write_by_dnd" type="xs:string" use="optional
" />
        <xs:attribute name="voicemail_remove_user_intro" type="xs:string" use="optional"
/>
        <xs:attribute name="voicemail_write_user_intro" type="xs:string" use="optional" /
>
        <xs:attribute name="voicemail_say_message_after_beep" type="xs:string" use="optio
nal" />
        <xs:attribute name="voicemail_reach_max_duration" type="xs:string" use="optional"
/>

        <xs:attribute name="monday" type="xs:string" use="optional" />
        <xs:attribute name="tuesday" type="xs:string" use="optional" />
        <xs:attribute name="wednesday" type="xs:string" use="optional" />
        <xs:attribute name="wednesday2" type="xs:string" use="optional" />
        <xs:attribute name="thursday" type="xs:string" use="optional" />
        <xs:attribute name="friday" type="xs:string" use="optional" />
        <xs:attribute name="friday2" type="xs:string" use="optional" />
        <xs:attribute name="saturday" type="xs:string" use="optional" />
        <xs:attribute name="saturday2" type="xs:string" use="optional" />
        <xs:attribute name="sunday" type="xs:string" use="optional" />
        <xs:attribute name="alarm_on" type="xs:string" use="optional" />
        <xs:attribute name="alarm_off" type="xs:string" use="optional" />
        <xs:attribute name="alarm_today_off" type="xs:string" use="optional" />
        <xs:attribute name="nearest1" type="xs:string" use="optional" />
        <xs:attribute name="nearest2" type="xs:string" use="optional" />
        <xs:attribute name="nearest3" type="xs:string" use="optional" />
        <xs:attribute name="nearest4" type="xs:string" use="optional" />
        <xs:attribute name="every1" type="xs:string" use="optional" />
        <xs:attribute name="every2" type="xs:string" use="optional" />
        <xs:attribute name="every3" type="xs:string" use="optional" />
        <xs:attribute name="every4" type="xs:string" use="optional" />
        <xs:attribute name="call_forwarding" type="xs:string" use="optional" />
        <xs:attribute name="from" type="xs:string" use="optional" />
        <xs:attribute name="untill" type="xs:string" use="optional" />
        <xs:attribute name="activated" type="xs:string" use="optional" />
        <xs:attribute name="second1" type="xs:string" use="optional" />
        <xs:attribute name="second2" type="xs:string" use="optional" />
        <xs:attribute name="second3" type="xs:string" use="optional" />
        <xs:attribute name="number_is_hidden" type="xs:string" use="optional" />
        <xs:attribute name="autoreodial_with_cb_inactive" type="xs:string" use="optional"
/>

        <xs:attribute name="autoreodial_with_cb_deactivated" type="xs:string" use="optiona
l" />

        <xs:attribute name="ms_ringback_second" type="xs:string" use="optional" />
        <xs:attribute name="ms_release_busy" type="xs:string" use="optional" />
        <xs:attribute name="ms_preemption_busy" type="xs:string" use="optional" />

```

```

/>
<xs:attribute name="call_recording_notification" type="xs:string" use="optional"
/>
<xs:attribute name="call_recording_announcement" type="xs:string" use="optional"
/>
<xs:attribute name="cc_supervising_mode_selected" type="xs:string" use="optional"
/>
<xs:attribute name="cc_supervising_mode_observing" type="xs:string" use="optional
" />
<xs:attribute name="cc_supervising_mode_consult" type="xs:string" use="optional"
/>
<xs:attribute name="cc_supervising_mode_conference" type="xs:string" use="optiona
1" />
<xs:attribute name="teleconference_greeting" type="xs:string" use="optional" />
<xs:attribute name="teleconference_farewell" type="xs:string" use="optional" />
<xs:attribute name="teleconference_join" type="xs:string" use="optional" />
<xs:attribute name="teleconference_leave" type="xs:string" use="optional" />
<xs:attribute name="conference_greeting" type="xs:string" use="optional" />
<xs:attribute name="conference_farewell" type="xs:string" use="optional" />
<xs:attribute name="conference_full" type="xs:string" use="optional" />
<xs:attribute name="conference_restriction" type="xs:string" use="optional" />
<xs:attribute name="conference_already_exists" type="xs:string" use="optional" />
<xs:attribute name="conference_not_exists" type="xs:string" use="optional" />
<xs:attribute name="conference_join" type="xs:string" use="optional" />
<xs:attribute name="conference_leave" type="xs:string" use="optional" />
<xs:attribute name="voicemail_message_from_hidden_number" type="xs:string" use="o
ptional" />
<xs:attribute name="zone_page_ready_tone" type="xs:string" use="optional" />
<xs:attribute name="exact_time" type="xs:string" use="optional" />
<xs:attribute name="mailbox_empty" type="xs:string" use="optional" />
<xs:attribute name="ringback" type="xs:string" use="optional" />
<xs:attribute name="subscriber_do_not_disturb" type="xs:string" use="optional" />
</xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

```

```

        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/core_ai_set

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_ai_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="coreAIOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="coreAIOptionsType" wait_time="system://sounds/ai_wait_time.wav"
      />
    </domain>
  </domain_options>
</request>
</in>

```

Ответ: 200


```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_ai_se
t.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="wait_time"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Hc_core_messages - Управление настройками сообщений

- [Hc_core_messages_clean](#)
- [Hc_core_messages_info](#)
- [Hc_core_messages_set](#)

Hc_core_messages_clean

Команда для восстановления настроек сообщений.

Аналог команды в Soco:

domain/<DOMAIN>/core/<core>/messages/clean

Шаблон URL-запроса:

http://host:port/commands/core_messages_clean

Коды ответа

200 - успех;

404 - ошибка.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean Core domain properties (text messages auto-informers).
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/core_messages_clean

```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_messa
ges_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="ss_activate_error"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_messa
ges_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="ss_activate_error"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Ис_core_messages_info

Команда предназначена для просмотра параметров сообщений

Аналог команды в Soap:

domain/<DOMAIN>/core/<core>/messages/info

Шаблон URL-запроса:

http://host:port/commands/core_messages_info

Коды ответа

200 - успех;

404 - ошибка.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show Core domain properties (text messages auto-informers).

      The list of the messages settings splitted by semicolon.
      language - message language;
      text - message text.

      Sample:
      error=language:ru,text:Активирован;
      error=language:en,text:Activated

      Copyright (c) 2018, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="coreMessagesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="error" type="xs:string" use="optional" />
        <xs:attribute name="ss_activated" type="xs:string" use="optional" />
        <xs:attribute name="ss_changed" type="xs:string" use="optional" />
        <xs:attribute name="ss_deactivated" type="xs:string" use="optional" />
        <xs:attribute name="ss_deactivated_all" type="xs:string" use="optional" />
        <xs:attribute name="ss_activate_error" type="xs:string" use="optional" />
        <xs:attribute name="ss_already_activated" type="xs:string" use="optional" />
        <xs:attribute name="ss_deactivate_error" type="xs:string" use="optional" />
        <xs:attribute name="ss_test_error" type="xs:string" use="optional" />
        <xs:attribute name="pin_incorrect" type="xs:string" use="optional" />
        <xs:attribute name="alarm_on" type="xs:string" use="optional" />
        <xs:attribute name="call_forwarding" type="xs:string" use="optional" />
        <xs:attribute name="wait_time" type="xs:string" use="optional" />
        <xs:attribute name="my_number" type="xs:string" use="optional" />
        <xs:attribute name="ivr_block_fax_subject" type="xs:string" use="optional" />
        <xs:attribute name="ivr_block_fax_body" type="xs:string" use="optional" />
        <xs:attribute name="ivr_block_fax_error_subject" type="xs:string" use="optional"
/>
        <xs:attribute name="ivr_block_fax_error_body" type="xs:string" use="optional" />
        <xs:attribute name="cc_queue_lost_call_subject" type="xs:string" use="optional" /
>
        <xs:attribute name="cc_queue_lost_call_body" type="xs:string" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoInType" minOccurs="
1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/core_messages_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_messa
ges_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_messages_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="coreMessagesOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="coreMessagesOptionsType" wait_time="language:default,text:Ожидание;language:ru,text:Ожидание;language:en,text:Waiting" ss_test_error="language:default,text:Ошибка тестирования;language:ru,text:Ошибка тестирования;language:en,text:Test error"
        ss_deactivated_all="language:default,text:Деактивирован;language:ru,text:Деактивирован;language:en,text:Deactivated"
        ss_deactivated="language:default,text:Деактивирован;language:ru,text:Деактивирован;language:en,text:Deactivated"
        ss_deactivate_error="language:default,text:Ошибка деактивации;language:ru,text:Ошибка деактивации;language:en,text:Deactivate error"
        ss_changed="language:default,text:Изменен;language:ru,text:Изменен;language:en,text:Changed"
        ss_already_activated="language:default,text:Услуга уже активирована;language:ru,text:Услуга уже активирована;language:en,text:Already activated"
        ss_activated="language:default,text:Активирован;language:ru,text:Активирован;language:en,text:Activated"
        ss_activate_error="language:default,text:Ошибка активации;language:ru,text:Ошибка активации;language:en,text:Activate error"
        pin_incorrect="language:default,text:Неверный PIN;language:ru,text:Неверный PIN;language:en,text:PIN incorrect"
        my_number="language:default,text:Мой номер {};language:ru,text:Мой номер {};language:en,text:Current number {}"
        ivr_block_fax_subject="language:default,text:Факс от {};language:ru,text:Факс от {};language:en,text:Fax from {}"
        ivr_block_fax_error_subject="language:default,text:Не удалось принять факс от {}({});language:ru,text:Не удалось принять факс от {}({});language:en,text:Cannot receive fax from {}({})"
        ivr_block_fax_error_body="language:default,text:Время: {} Отправитель: {}({}) Получатель: {}({});language:ru,text:Время: {} Отправитель: {}({}) Получатель: {}({});language:en,text:Time: {} Sender: {}({}) Receiver: {}({})"
        ivr_block_fax_body="language:default,text:Получено: {} Отправитель: {}({}) Получатель: {}({});language:ru,text:Получено: {} Отправитель: {}({}) Получатель: {}({});language:en,text:Received: {} Sender: {}({}) Receiver: {}({})"
        error="language:default,text:Ошибка;language:ru,text:Ошибка;language:en,text:Error"
        cc_queue_lost_call_subject="language:default,text:Пропущенный вызов от абонента;language:ru,text:Пропущенный вызов от абонента;language:en,text:Call missed"
        cc_queue_lost_call_body="language:default,text:Абонент: {} Время звонка: {} Время ожидания: {} сек. Количество попыток распределения: {} Операторы не ответившие на вызов;;language:ru,text:Абонент: {} Время звонка: {} Время ожидания: {} сек. Количество попыток распределения: {} Операторы не ответившие на вызов;;language:en,text:Subscriber: {} Add time: {} Waiting time: {} sec. Distributions count: {} Operators who have missed a call:"
        call_forwarding="language:default,text:Переадресация на;language:ru,text:Переадресация на;language:en,text:Call forwarding on"
        alarm_on="language:default,text:Будильник установлен;language:ru,text:Будильник установлен;language:en,text:Alarm on"/>
      </domain>
    </domain_options>
  </result>
</out>

```

[Hc_core_messages_set](#)

Команда предназначена для установки параметров сообщений

Аналог команды в Soap:

domain/<DOMAIN>/core/<core>/messages/set

Шаблон URL-запроса:

http://host:port/commands/core_messages_set

Коды ответа

200 - успех;

404 - ошибка.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Change Core domain properties (auto-informers).

      The list of the messages settings splitted by semicolon.
      language - message language;
      text - message text.

      Sample:
      language:ru,text:Активирован;
      language:en,text:Activated

      Copyright (c) 2018, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="coreMessagesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="error" type="xs:string" use="optional" />
        <xs:attribute name="ss_activated" type="xs:string" use="optional" />
        <xs:attribute name="ss_changed" type="xs:string" use="optional" />
        <xs:attribute name="ss_deactivated" type="xs:string" use="optional" />
        <xs:attribute name="ss_deactivated_all" type="xs:string" use="optional" />
        <xs:attribute name="ss_activate_error" type="xs:string" use="optional" />
        <xs:attribute name="ss_already_activated" type="xs:string" use="optional" />
        <xs:attribute name="ss_deactivate_error" type="xs:string" use="optional" />
        <xs:attribute name="ss_test_error" type="xs:string" use="optional" />
        <xs:attribute name="pin_incorrect" type="xs:string" use="optional" />
        <xs:attribute name="alarm_on" type="xs:string" use="optional" />
        <xs:attribute name="call_forwarding" type="xs:string" use="optional" />
        <xs:attribute name="wait_time" type="xs:string" use="optional" />
        <xs:attribute name="my_number" type="xs:string" use="optional" />
        <xs:attribute name="ivr_block_fax_subject" type="xs:string" use="optional" />
        <xs:attribute name="ivr_block_fax_body" type="xs:string" use="optional" />
        <xs:attribute name="ivr_block_fax_error_subject" type="xs:string" use="optional" />
        <xs:attribute name="ivr_block_fax_error_body" type="xs:string" use="optional" />
        <xs:attribute name="cc_queue_lost_call_subject" type="xs:string" use="optional" />
        <xs:attribute name="cc_queue_lost_call_body" type="xs:string" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```



```

</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/core_messages_set

```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_messa
ges_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="coreMessagesOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="coreMessagesOptionsType" ss_activate_error="language:default,te
xt:Ошибка активации;language:ru,text:Ошибка активации;language:en,text:Activate error"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_messa
ges_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="ss_activate_error"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Hc_core_timers - Управление таймерами виртуальной АТС

- [Hc_core_timers_clean](#)
- [Hc_core_timers_info](#)
- [Hc_core_timers_set](#)

Hc_core_timers_clean

Команда сброса настроек таймеров указанной виртуальной АТС.

Аналог команды в CoCon:

domain/<DOMAIN>/timers/core/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/core_timers_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="core_timers_clean">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean Core timer properties
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"
                minOccurs="1" maxOccurs="1" />
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanOutType"
                minOccurs="1" maxOccurs="1" />
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- out -->

</xs:schema>

```

Пример:

http://192.168.1.21:9999/commands/core_timers_clean

Запрос:

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="core_timers_clean.xsd">
  <request>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <option name="conversation_timeout"/>
        <option name="o_response_timeout"/>
        <option name="suspend_timeout_t2"/>
        <option name="suspend_timeout_t38"/>
        <option name="suspend_timeout_t6"/>
        <option name="release_ack_timeout"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_timer
s_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="conversation_timeout"/>
        <success option="release_ack_timeout"/>
        <success option="suspend_timeout_t2"/>
        <success option="suspend_timeout_t38"/>
        <success option="suspend_timeout_t6"/>
        <error option="o_response_timeout" reason="{error,not_found}"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Hc_core_timers_info

Команда просмотра свойств таймеров указанной виртуальной АТС.

Аналог команды в CoCon:

domain/<DOMAIN>/timers/core/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/core_timers_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="core_timers_info">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show Core timers properties
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <xs:complexType name="octTimersOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="abonent_manager_statistic_boom_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="cc_queue_remember_choice_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="fill_seizure_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="release_ack_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="trunk_manager_cache_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="voicemail_read_message_cleanup_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="conversation_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="no_answer_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="o_no_answer_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="response_ack_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="response_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="router_resp_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="suspend_timeout_t38" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="suspend_timeout_t6" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="suspend_timeout_t2" type="nonEmptyTokenType" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">

```

```

<xs:complexType>
  <xs:all>
    <xs:element name="request" minOccurs="1" maxOccurs="1">
      <xs:complexType>
        <xs:all>
          <xs:element name="domain_options" type="domainInfoInType"
            minOccurs="1" maxOccurs="1" />
        </xs:all>
      </xs:complexType>
    </xs:element>
  </xs:all>
</xs:complexType>
</xs:element>
<!-- in -->

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoOutType"
              minOccurs="1" maxOccurs="1" />
          </xs:all>
        </xs:complexType>
      </xs:element>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
    </xs:choice>
  </xs:complexType>
</xs:element>
<!-- out -->

</xs:schema>

```

Пример:

http://192.168.1.21:9999/commands/core_timers_info

Запрос значений всех доступных свойств:

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_timer
s_info.xsd">
  <request>
    <domain_options domain="biysk.local" />
  </request>
</in>

```


Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_timer
s_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="octTimersOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="octTimersOptionsType" voicemail_read_message_cleanup_timeout="8
6400000" trunk_manager_cache_timeout="60000" suspend_timeout_t6="60000" suspend_timeout_t38="9000
0" suspend_timeout_t2="180000" router_resp_timeout="5000" response_timeout="25000"
response_ack_timeout="6000" release_ack_timeout="5000" no_answer_timeout="60000"
conversation_timeout="3600000" cc_queue_remember_choice_timeout="3600000"
abonent_manager_statistic_boom_timeout="300000"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Нс_core_timers_set

Команда изменения свойств таймера указанной виртуальной АТС.

Аналог команды в CoCon:

domain/<DOMAIN>/timers/core/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/core_timers_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="core_timers_set">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Change Core timers properties
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <xs:complexType name="octTimersOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="abonent_manager_statistic_boom_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="cc_queue_remember_choice_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="fill_seizure_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="release_ack_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="trunk_manager_cache_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="voicemail_read_message_cleanup_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="conversation_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="no_answer_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="o_no_answer_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="response_ack_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="response_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="router_resp_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="suspend_timeout_t38" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="suspend_timeout_t6" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="suspend_timeout_t2" type="nonEmptyTokenType" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainSetInType"
                minOccurs="1" maxOccurs="1" />
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

        </xs:element>
    </xs:all>
</xs:complexType>
</xs:element>
<!-- in -->

<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainSetOutType"
                            minOccurs="1" maxOccurs="1" />
                    </xs:all>
                </xs:complexType>
            </xs:element>
            <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
        </xs:choice>
    </xs:complexType>
</xs:element>
<!-- out -->

</xs:schema>

```

Пример:

http://192.168.1.21:9999/commands/core_timers_set

Запрос:

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_timer
s_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="octTimersOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="octTimersOptionsType" conversation_timeout="9"
no_answer_timeout="60" suspend_timeout_t38="12" suspend_timeout_t6="disabled" suspend_timeout_t2="
15" router_resp_timeout="6"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="core_timer
s_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="conversation_timeout"/>
        <success option="no_answer_timeout"/>
        <success option="router_resp_timeout"/>
        <success option="suspend_timeout_t6"/>
        <error option="suspend_timeout_t2" reason="{invalid_value,15}"/>
        <error option="suspend_timeout_t38" reason="{invalid_value,12}"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Hc_custom_cb - Управление виджетом сервиса "custom callback"

- [Hc_custom_cb_new](#)
- [Hc_custom_cb_set](#)
- [Hc_custom_cb_list](#)
- [Hc_custom_cb_remove](#)

Hc_custom_cb_new

Команда по созданию callback-виджета.

Аналог команды в CoCon:

`/domain/<DOMAIN>/custom_cb/new`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://server:9999/commands/custom_cb_new`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="common_types.xsd"/>
  <!-- Types -->
  <xs:complexType name="widgetType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="from_number" type="nonEmptyString" use="required"/>
    <xs:attribute name="name" type="nonEmptyString" use="required"/>
    <xs:attribute name="scenario" type="nonEmptyString" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="widget" type="widgetType"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="responseType">
    <xs:attribute name="error" type="xs:string" use="optional"/>
    <xs:attribute name="id" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

http://192.168.1.21:9999/commands/custom_cb_new

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_custom_cb_new.xsd">
  <request domain="nick">
    <widget from_number="1563" name="test_ccb" scenario="default" description="abc"/>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_custom_cb_new.xsd">
  <response id="0655e332b2bf7dd0"/>
</out>
```

Нс_custom_cb_set

Команда по редактированию callback-виджета.

Аналог команды в CoCon:

/domain/<DOMAIN>/custom_cb/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/custom_cb_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="common_types.xsd"/>
  <!-- Types -->
  <xs:complexType name="widgetType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="nonEmptyString" use="required"/>
    <xs:attribute name="from_number" type="nonEmptyString" use="optional"/>
    <xs:attribute name="name" type="nonEmptyString" use="optional"/>
    <xs:attribute name="scenario" type="nonEmptyString" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="widget" type="widgetType"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="responseType">
    <xs:attribute name="error" type="xs:string" use="optional"/>
    <xs:attribute name="id" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

http://192.168.1.21:9999/commands/custom_cb_set

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_custom_cb_set.xsd">
  <request domain="nick">
    <widget id="0655e2e5c48fded2">
      <property name="key1" value="value111" />
      <property name="key2" value="value22" />
    </widget>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_custom_cb_set.xsd">
  <response id="0655e2e5c48fded2"/>
</out>
```

hc_custom_cb_list

Команда получения списка callback-виджетов.

Аналог команды в CoCon:

/domain/<DOMAIN>/custom_cb/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/custom_cb_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="common_types.xsd"/>
  <!-- Types -->
  <xs:complexType name="widgetType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="from_number" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="scenario" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="widget" type="widgetType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/custom_cb_list

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_custom_cb_list.xsd">
  <request domain="biysk.local"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_custom_cb_list.xsd">
  <response>
    <widget id="06220252f0137bea" from_number="240473" name="new_73"
scenario="0622071a81711369" description="new_wid"/>
    <widget id="062229da8ce73a45" from_number="240471" name="check_widget"
scenario="default_custom_callback" description="abc">
      <property name="key1" value="value111"/>
      <property name="key2" value="value22"/>
    </widget>
  </response>
</out>
```

[Hc_custom_cb_remove](#)

Команда удаления callback-виджета.

Аналог команды в CoCon:

`/domain/<DOMAIN>/custom_cb/delete/`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://server:9999/commands/custom_cb_remove`

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:complexType name="widgetType">
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="widget" type="widgetType"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="responseType">
    <xs:attribute name="error" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/custom_cb_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_custom_cb_remove.xsd">
  <request domain="biysk.local">
    <widget id="062229da8ce73a45" />
  </request>
</in>
```

Ответ:

В случае успеха будет 201 ответ, в случае ошибки 404 + xml с описанием ошибки:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_custom_cb_remove.xsd">
  <response error="not_found"/>
</out>
```

Hc_dialer_campaign - Управление кампаниями дозвона

- [Hc_dialer_campaign_declare](#)
- [Hc_dialer_campaign_info](#)
- [Hc_dialer_campaign_action](#)
- [Hc_dialer_campaign_list](#)
- [Hc_dialer_campaign_start](#)

В данном разделе описаны команды для управления кампаниями дозвона.

[Hc_dialer_campaign_declare](#)

Команда предназначена для создания кампании дозвона.

Аналог команды в CoCon:

/domain/<DOMAIN>/ss/dialer/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/dialer_campaign_declare

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема dialer_common.xsd:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:complexType name="numbersType">
    <xs:sequence>
      <xs:element name="number" type="numberType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="propertyDescriptionsType">
    <xs:sequence>
      <xs:element name="property" type="propertyDescriptionType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="scriptsType">
    <xs:sequence>
      <xs:element name="script" type="scriptType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="operatorNumbersType">
    <xs:sequence>
      <xs:element name="number" type="operatorNumberType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="scriptType">
    <xs:sequence>
      <xs:element name="block" type="blockType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="blockType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="output" type="xs:string" use="required"/>
    <xs:attribute name="result" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="numberType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="digits" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="operatorNumberType">

```

```
<xs:attribute name="digits" type="xs:string"/>
</xs:complexType>

<xs:complexType name="propertyType">
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="value" type="xs:string" use="required"/>
</xs:complexType>

<xs:complexType name="propertyDescriptionType">
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="data_name" type="xs:string"/>
  <xs:attribute name="report_name" type="xs:string"/>
</xs:complexType>
</xs:schema>
```

XSD-схема hc_dialer_campaign_declare.xsd:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="dialer_common.xsd"/>
  <!-- campaignType -->
  <xs:complexType name="campaignType">
    <xs:sequence>
      <xs:element name="numbers" type="numbersType" minOccurs="0"/>
      <xs:element name="operator_numbers" type="operatorNumbersType" minOccurs="0"/>
      <xs:element name="property_descriptions" type="propertyDescriptionsType" minOccurs="0"
"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="template" type="xs:string" use="required"/>
    <xs:attribute name="call_success_by_variable" type="xs:string"/>
    <xs:attribute name="start_type" type="xs:string" use="required"/>
    <xs:attribute name="lines_limit" type="xs:nonNegativeInteger"/>
    <xs:attribute name="calls_limit" type="xs:nonNegativeInteger"/>
    <xs:attribute name="recall_timeout" type="xs:nonNegativeInteger"/>
    <xs:attribute name="no_answer_timeout" type="xs:nonNegativeInteger"/>
    <xs:attribute name="priority" type="xs:nonNegativeInteger"/>
    <xs:attribute name="save_all_attempts" type="xs:boolean"/>
    <xs:attribute name="time_interval" type="xs:string"/>
    <xs:attribute name="min_duration" type="xs:nonNegativeInteger"/>
    <xs:attribute name="group" type="xs:string"/>
    <xs:attribute name="shuffle" type="xs:boolean"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="campaign" type="campaignType"/>
            </xs:all>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:complexType name="responseType">
    <xs:attribute name="campaign_id" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="out">
    <xs:complexType>

```



```

    <xs:all>
      <xs:element name="response" type="responseType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/dialer_campaign_declare

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_
campaign_declare.xsd">
  <request domain="biysk.local">
    <campaign name="info1" template="info4lab" start_type="now" lines_limit="2" calls_limit="
2" recall_timeout="60" time_interval="09:00-18:00" min_duration="5" shuffle="true"/>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_
campaign_declare.xsd">
  <response campaign_id="2"/>
</out>

```

hc_dialer_campaign_info

Команда предназначена для просмотра информации о кампании дозвона.

Аналог команды в CoCon:

/domain/<DOMAIN>/ss/dialer/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/dialer_campaign_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="dialer_common.xsd"/>
  <!-- campaignType -->
  <xs:complexType name="campaignType">
    <xs:sequence>
      <xs:element name="call" type="callType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="campaign_id" type="xs:string"/>
    <xs:attribute name="name" type="xs:string"/>
    <xs:attribute name="lines_limit" type="xs:nonNegativeInteger"/>
    <xs:attribute name="calls_limit" type="xs:nonNegativeInteger"/>
    <xs:attribute name="recall_timeout" type="xs:nonNegativeInteger"/>
    <xs:attribute name="time_interval" type="xs:string"/>
    <xs:attribute name="min_duration" type="xs:nonNegativeInteger"/>
    <xs:attribute name="start_time" type="xs:string"/>
    <xs:attribute name="update_time" type="xs:string"/>
    <xs:attribute name="status" type="xs:string"/>
    <xs:attribute name="properties" type="xs:string"/>
  </xs:complexType>
  <xs:complexType name="callType">
    <xs:attribute name="number" type="xs:string"/>
    <xs:attribute name="start_time" type="xs:string"/>
    <xs:attribute name="answer_time" type="xs:string"/>
    <xs:attribute name="release_time" type="xs:string"/>
    <xs:attribute name="retry_number" type="xs:string"/>
    <xs:attribute name="status" type="xs:string"/>
    <xs:attribute name="result" type="xs:string"/>
    <xs:attribute name="ivr_variables" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- campaignsType -->
  <xs:complexType name="campaignsType">
    <xs:sequence>
      <xs:element name="campaign" type="campaignType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="campaign_id" type="xs:nonNegativeInteger" use="required"/>
            <xs:attribute name="active" type="xs:boolean" default="false"/>
            <xs:attribute name="limit" type="xs:nonNegativeInteger" default="50"/>
            <xs:attribute name="offset" type="xs:nonNegativeInteger" default="0"/>
            <xs:attribute name="show_options" type="xs:boolean" default="false"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

        <xs:attribute name="show_ivr_variables" type="xs:boolean" default="false"
/>
        </xs:complexType>
    </xs:element>
</xs:all>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:all>
            <xs:element name="campaign" type="campaignType"/>
        </xs:all>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/dialer_campaign_info

```

<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_diale
r_campaign_info.xsd">
  <request domain="biysk.local" campaign_id="2" limit="5"/>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_dialer_campaign_info.xsd">
  <campaign>
    <call number="240510" start_time="11.09.2019 05:57:54" answer_time="11.09.2019 05:57:56"
    release_time="11.09.2019 05:58:30" retry_number="1" status="success" result=""/>
    <call number="240511" start_time="11.09.2019 05:57:54" answer_time="11.09.2019 05:57:56"
    release_time="11.09.2019 05:58:31" retry_number="1" status="success" result=""/>
    <call number="416370" start_time="11.09.2019 05:58:30" answer_time="11.09.2019 05:58:33"
    release_time="11.09.2019 05:59:04" retry_number="1" status="success" result=""/>
    <call number="416977" start_time="11.09.2019 05:58:31" answer_time="11.09.2019 05:58:42"
    release_time="11.09.2019 05:58:54" retry_number="1" status="success" result=""/>
  </campaign>
</out>

```

Нс_dialer_campaign_action

Команда предназначена для задания команд кампаниям дозвона.

Аналог команды в CoCon:

`/domain/<DOMAIN>/ss/dialer/<remove|stop|pause|unpause>`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/dialer_campaign_action`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- actionType -->
  <xs:complexType name="actionType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="campaign_id" type="xs:nonNegativeInteger"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="action" type="actionType"/>
            </xs:all>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:complexType name="responseType">
    <xs:attribute name="result" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Примеры

Запрос:

http://192.168.1.21:9999/commands/dialer_campaign_action

Запрос (pause):

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_
campaign_action.xsd">
  <request domain="biysk.local">
    <action name="pause" campaign_id="5"/>
  </request>
</in>
```

Запрос (unpause):

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_
campaign_action.xsd">
  <request domain="biysk.local">
    <action name="unpause" campaign_id="7"/>
  </request>
</in>
```

Запрос (stop):

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_
campaign_action.xsd">
  <request domain="biysk.local">
    <action name="stop" campaign_id="6"/>
  </request>
</in>
```

Запрос (remove):

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_
campaign_action.xsd">
  <request domain="biysk.local">
    <action name="remove" campaign_id="5"/>
  </request>
</in>
```

Ответ (pause):

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_campaign_action.xsd">
  <response result="paused"/>
</out>
```

ОТВЕТ (unpause):

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_campaign_action.xsd">
  <response result="unpaused"/>
</out>
```

ОТВЕТ (stop):

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_campaign_action.xsd">
  <response result="stopped"/>
</out>
```

ОТВЕТ (remove):

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_campaign_action.xsd">
  <response result="removed"/>
</out>
```

hc_dialer_campaign_list

Команда предназначена для просмотра списка кампаний дозвона.

Аналог команды в CoCon:

```
/domain/<DOMAIN>/ss/dialer/list
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/dialer_campaign_list

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="dialer_common.xsd"/>
  <!-- campaignType -->
  <xs:complexType name="campaignType">
    <xs:attribute name="campaign_id" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="lines_limit" type="xs:string" use="required"/>
    <xs:attribute name="calls_limit" type="xs:string" use="required"/>
    <xs:attribute name="recall_timeout" type="xs:string" use="required"/>
    <xs:attribute name="time_interval" type="xs:string" use="required"/>
    <xs:attribute name="min_duration" type="xs:string" use="required"/>
    <xs:attribute name="start_time" type="xs:string" use="required"/>
    <xs:attribute name="update_time" type="xs:string" use="required"/>
    <xs:attribute name="status" type="xs:string" use="required"/>
    <xs:attribute name="dialed_numbers" type="xs:string" use="optional"/>
    <xs:attribute name="total_numbers" type="xs:string" use="optional"/>
    <xs:attribute name="percent_complete" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- campaignsType -->
  <xs:complexType name="campaignsType">
    <xs:sequence>
      <xs:element name="campaign" type="campaignType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="filter" type="filterType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="valueType">
    <xs:attribute name="operator" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="filterType">
    <xs:sequence>
      <xs:element name="property" type="valueType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="property" type="xs:string" use="required"/>
    <xs:attribute name="delim" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="result" type="campaignsType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/dialer_campaign_list

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_diale
r_campaign_list.xsd">
  <request domain="biysk.local"/>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_
campaign_list.xsd">
  <result>
    <campaign campaign_id="1" name="dialer_test_2019_09_11_12_54_15" lines_limit="2"
calls_limit="1" recall_timeout="60" time_interval="09:00-18:00" min_duration="5" start_time="11.0
9.2019 05:54:29" update_time="11.09.2019 05:54:47" status="finished" dialed_numbers="4"
total_numbers="4" percent_complete="100.00"/>
    <campaign campaign_id="2" name="dialer_test_2019_09_11_12_57_49" lines_limit="2"
calls_limit="1" recall_timeout="60" time_interval="09:00-18:00" min_duration="5" start_time="11.0
9.2019 05:57:54" update_time="11.09.2019 05:59:05" status="finished" dialed_numbers="4"
total_numbers="4" percent_complete="100.00"/>
  </result>
</out>

```

hc_dialer_campaign_start

Команда предназначена для запуска кампании дозвона.

Аналог команды в CoCon:

/domain/<DOMAIN>/ss/dialer/start

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/dialer_campaign_start

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- actionType -->
  <xs:complexType name="startType">
    <xs:attribute name="campaign_id" type="xs:nonNegativeInteger" use="required"/>
    <xs:attribute name="start_type" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="start" type="startType"/>
            </xs:all>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:complexType name="responseType">
    <xs:attribute name="campaign_id" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Примеры

Запрос:

http://192.168.1.21:9999/commands/dialer_campaign_start

Запрос (start_type=now):

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_campaign_start.xsd">
  <request domain="biysk.local">
    <start campaign_id="1" start_type="now"/>
  </request>
</in>
```

Запрос (start_type=manual):

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_campaign_start.xsd">
  <request domain="biysk.local">
    <start campaign_id="2" start_type="manual"/>
  </request>
</in>
```

Запрос (start_type=datetime):

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_campaign_start.xsd">
  <request domain="biysk.local">
    <start campaign_id="2" start_type="17.11.2019 21:31:10"/>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_dialer_campaign_start.xsd">
  <response campaign_id="2"/>
</out>
```

Hc_dialer_db_cluster_options - Управление параметрами подключения в БД подсистемы автообзвонщика

- [Hc_dialer_db_cluster_options_clean](#)
- [Hc_dialer_db_cluster_options_info](#)
- [Hc_dialer_db_cluster_options_set](#)

Нс_dialer_db_cluster_options_clean

Команда сброса параметров подключения к БД подсистемы автообзвонщика.

Аналог команды в CoCon:

`cluster/core/<CORE>/dialer/db/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/dialer_db_cluster_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean dialer cluster options.
      Usage: http://server:9999/commands/dialer_db_cluster_options_clean
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterCleanOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/dialer_db_cluster_options_clean


```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="dialer_db_cluster_options_clean.xsd">
  <request>
    <cluster_options>
      <cluster name="core1">
        <option name="mysql_host"/>
        <option name="mysql_port"/>
      </cluster>
    </cluster_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="dialer_db_cluster_options_clean.xsd">
  <result>
    <cluster_options>
      <cluster name="core1">
        <success option="mysql_port"/>
        <success option="mysql_host"/>
      </cluster>
    </cluster_options>
  </result>
</out>

```

[Hc_dialer_db_cluster_options_info](#)

Команда просмотра параметров подключения к БД подсистемы автообзвонщика.

Аналог команды в CoCon:

cluster/core/<CORE>/dialer/db/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/dialer_db_cluster_options_info

Код HTTP-ответа:

- 200 – в случае успеха;

- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get options of dialer node.
      Usage: http://server:9999/commands/dialer\_db\_cluster\_options\_info
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <!-- dialerDbClusterOptionsType -->
  <xs:complexType name="dialerDbClusterOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="mysql_database" type="nonEmptyNormalizedStringType" use="optional" />
        <xs:attribute name="mysql_enabled" type="booleanType" use="optional" />
        <xs:attribute name="mysql_host" type="nonEmptyNormalizedStringType" use="optional" />
        <xs:attribute name="mysql_password" type="nonEmptyNormalizedStringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              In case of strong security restrictions password may be shown as
              *****
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="mysql_port" type="inetPortType" use="optional" />
        <xs:attribute name="mysql_user" type="nonEmptyNormalizedStringType" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterInfoInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:all minOccurs="1" maxOccurs="1">
            <xs:element name="cluster_options" type="clusterInfoOutType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/dialer_db_cluster_options_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request>
    <cluster_options cluster="core1"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="dialer_db_cluster_options_info.xsd">
  <result>
    <cluster_options>
      <cluster name="core1">
        <options xs:type="dialerDbClusterOptionsType" mysql_database="ecss_dialer_db"
mysql_enabled="true" mysql_host="dialer.mysql.ecss" mysql_password="*****" mysql_port="3306"
mysql_user="dialer_db"/>
      </cluster>
    </cluster_options>
  </result>
</out>
```

hc_dialer_db_cluster_options_set

Изменение параметров подключения к БД подсистемы автообзвонщика.

Аналог команды в CoCon:

cluster/core/<CORE>/dialer/db/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/dialer_db_cluster_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set dialer cluster options.
      Usage: http://server:9999/commands/dialer_db_cluster_options_set
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <xs:complexType name="dialerDbClusterOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="mysql_database" type="nonEmptyNormalizedStringType"
use="optional" />
        <xs:attribute name="mysql_enabled" type="booleanType" use="optional" />
        <xs:attribute name="mysql_host" type="nonEmptyNormalizedStringType"
use="optional" />
        <xs:attribute name="mysql_password" type="nonEmptyNormalizedStringType"
use="optional" />
        <xs:attribute name="mysql_port" type="inetPortType" use="optional" />
        <xs:attribute name="mysql_user" type="nonEmptyNormalizedStringType"
use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>

```

```

        <xs:all minOccurs="1" maxOccurs="1">
            <xs:element name="cluster_options" type="clusterSetOutType"/>
        </xs:all>
    </xs:complexType>
</xs:element>
<xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
</xs:choice>
</xs:complexType>
</xs:element>

</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/dialer_db_cluster_options_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="dialer_db_cluster_options_set.xsd">
  <request>
    <cluster_options>
      <cluster name="core1">
        <options mysql_host="192.168.1.71" mysql_port="3306"
xs:type="dialerDbClusterOptionsType"/>
      </cluster>
    </cluster_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="dialer_db_cluster_options_set.xsd">
  <result>
    <cluster_options>
      <cluster name="core1">
        <success option="mysql_host"/>
        <success option="mysql_port"/>
      </cluster>
    </cluster_options>
  </result>
</out>

```

Hc_dialer_report_number - Команда получения отчета по определенному номеру в автообзвоне

- [Hc_dialer_report_number_info](#)

[Hc_dialer_report_number_info](#)

Команда позволяет получить отчет по определенному номеру в автообзвоне.

Аналог команды в CoCon:

domain/<DOMAIN>/ss/dialer/report/number-info

Шаблон URL запроса

http://server:9999/commands/dialer_report_number_info

Коды ответа

200 - успех;

404 - ошибка.

XML-схема


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="dialer_common.xsd"/>
  <!-- Types -->
  <xs:simpleType name="dateTimeType">
    <xs:restriction base="xs:string">
      <xs:pattern value="\d{2}\.\d{2}\.\d{4}\s+\d{2}:\d{2}:\d{2}"/>
      <!-- DD.MM.YYYY HH:MM:SS -->
    </xs:restriction>
  </xs:simpleType>

  <!-- callType-->
  <xs:complexType name="callType">
    <xs:attribute name="number" type="xs:string"/>
    <xs:attribute name="campaign_name" type="xs:string"/>
    <xs:attribute name="start_time" type="xs:string"/>
    <xs:attribute name="answer_time" type="xs:string"/>
    <xs:attribute name="release_time" type="xs:string"/>
    <xs:attribute name="retry_number" type="xs:string"/>
    <xs:attribute name="status" type="xs:string"/>
    <xs:attribute name="result" type="xs:string"/>
    <xs:attribute name="ivr_variables" type="xs:string"/>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="number" type="xs:string" use="required"/>
            <xs:attribute name="tag" type="xs:base64Binary"/>
            <xs:attribute name="limit" type="xs:integer"/>
            <xs:attribute name="offset" type="xs:integer"/>
            <xs:attribute name="show_ivr_variables" type="xs:boolean"/>
            <xs:attribute name="from_date" type="dateTimeType"/>
            <xs:attribute name="to_date" type="dateTimeType"/>
            <xs:attribute name="ivr_variables" type="xs:string"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out-->
  <xs:complexType name="resultType">

```

```

<xs:sequence minOccurs="0">
  <xs:element name="call" type="callType" maxOccurs="unbounded"/>
</xs:sequence>
<xs:attribute name="tag" type="xs:base64Binary"/>
</xs:complexType>

<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="dialer_report_number_info.xsd">
  <request domain="biysk.local" number="240464" />
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_dialer_report_number_info.xsd">
  <result>
    <call campaign_name="market1_2020_02_13_08_25_05" number="240464" start_time="13.02.2020
01:25:37" answer_time="13.02.2020 01:25:40" release_time="13.02.2020 01:25:45" retry_number="1"
status="normal" result=""/>
    <call campaign_name="market1_2020_02_13_08_25_05" number="240464" start_time="13.02.2020
01:51:53" answer_time="13.02.2020 01:51:56" release_time="13.02.2020 01:51:57" retry_number="1"
status="normal" result=""/>
  </result>
</out>

```

В случае, если данные были отданы не все, в узле result добавится атрибут tag, который надо передать в последующем запросе для получения остальных данных.

hc_dialer_template - Управление шаблонами дозвона

- [Hc_dialer_template_declare](#)
- [Hc_dialer_template_info](#)
- [Hc_dialer_template_set](#)
- [Hc_dialer_template_list](#)
- [Hc_dialer_template_remove](#)

В данном разделе описаны команды для управления шаблонами дозвола.

[Hc_dialer_template_declare](#)

Команда предназначена для создания шаблона дозвола.

Аналог команды в CoCon:

`/domain/<DOMAIN>/ss/dialer/template/declare`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/dialer_template_declare

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
<xs:include schemaLocation="dialer_common.xsd"/>
<xs:complexType name="templateType">
<xs:all>
<xs:element name="numbers" type="numbersType" minOccurs="0"/>
<xs:element name="results" type="scriptsType" minOccurs="0"/>
<xs:element name="operator_numbers" type="operatorNumbersType" minOccurs="0"/>
<xs:element name="property_descriptions" type="propertyDescriptionsType" minOccurs="0"/>
</xs:all>
<xs:attribute name="name" type="xs:string" use="required"/>
<xs:attribute name="description" type="xs:string" use="required"/>
<xs:attribute name="script_id" type="xs:string" use="required"/>
<xs:attribute name="call_success_by_variable" type="xs:string"/>
<xs:attribute name="lines_limit" type="xs:nonNegativeInteger"/>
<xs:attribute name="calls_limit" type="xs:nonNegativeInteger"/>
<xs:attribute name="recall_timeout" type="xs:nonNegativeInteger"/>
<xs:attribute name="no_answer_timeout" type="xs:nonNegativeInteger"/>
<xs:attribute name="priority" type="xs:nonNegativeInteger"/>
<xs:attribute name="save_all_attempts" type="xs:boolean"/>
<xs:attribute name="time_interval" type="xs:string"/>
<xs:attribute name="min_duration" type="xs:nonNegativeInteger"/>
<xs:attribute name="group" type="xs:string"/>
<xs:attribute name="shuffle" type="xs:boolean"/>
</xs:complexType>
<!-- in -->
<xs:element name="in">
<xs:complexType>
<xs:all>
<xs:element name="request">
<xs:complexType>
<xs:all>
<xs:element name="template" type="templateType"/>
</xs:all>
<xs:attribute name="domain" type="xs:string" use="required"/>
</xs:complexType>
</xs:element>
</xs:all>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:complexType name="responseType">
<xs:attribute name="template_id" type="xs:string" use="required"/>
</xs:complexType>
<xs:element name="out">
<xs:complexType>
<xs:all>
<xs:element name="response" type="responseType"/>

```

```
</xs:all>  
</xs:complexType>  
</xs:element>  
</xs:schema>
```

Пример:

Запрос:

http://192.168.1.21:9999/commands/dialer_template_declare

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_dialer_template_declare.xsd">
  <request domain="biysk.local">
    <template name="test_name" description="Проверка unicode имен"
      script_id="063f86fccbeae10b" lines_limit="2" calls_limit="2" recall_timeout="60"
      time_interval="09:00-18:00" min_duration="5">
      <numbers>
        <number digits="4000">
          <property name="debt" value="500"/>
          <property name="name" value="vitaliy"/>
        </number>
        <number digits="4001">
          <property name="debt" value="500"/>
          <property name="name" value="Александр"/>
        </number>
        <number digits="4003">
          <property name="debt" value="500"/>
          <property name="name" value="maksim"/>
        </number>
        <number digits="4004">
          <property name="debt" value="500"/>
          <property name="name" value="alexey"/>
        </number>
      </numbers>
      <results>
        <script id="063f86fccbeae10b">
          <block name="ivr_3" output="3" result="Выборал3"/>
          <block name="play_4" output="true" result="Уведомлен"/>
        </script>
      </results>
      <operator_numbers>
        <number digits="4005"/>
        <number digits="4007"/>
      </operator_numbers>
      <property_descriptions>
        <property name="debt" data_name="debts" report_name="Задолженность"/>
        <property name="name" data_name="FIO"/>
      </property_descriptions>
    </template>
  </request>
</in>

```

Ответ:200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_dialer_template_declare.xsd">
  <response template_id="0643682845b69c28"/>
</out>
```

hc_dialer_template_info

Команда просмотра свойств шаблона дозвона.

Аналог команды в CoCon:

/domain/<DOMAIN>/ss/dialer/template/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/dialer_template_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="dialer_common.xsd"/>

  <xs:complexType name="templateReqType">
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="template" type="templateReqType"/>
            </xs:all>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:complexType name="templateType">
    <xs:all>
      <xs:element name="numbers" type="numbersType" minOccurs="0"/>
      <xs:element name="results" type="scriptsType" minOccurs="0"/>
      <xs:element name="operator_numbers" type="operatorNumbersType" minOccurs="0"/>
      <xs:element name="property_descriptions" type="propertyDescriptionsType"
minOccurs="0"/>
    </xs:all>
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
    <xs:attribute name="script" type="xs:string" use="required"/>
    <xs:attribute name="script_id" type="xs:string" use="required"/>
    <xs:attribute name="call_success_by_variable" type="xs:string"/>
    <xs:attribute name="lines_limit" type="xs:nonNegativeInteger"/>
    <xs:attribute name="calls_limit" type="xs:nonNegativeInteger"/>
    <xs:attribute name="recall_timeout" type="xs:nonNegativeInteger"/>
    <xs:attribute name="no_answer_timeout" type="xs:nonNegativeInteger"/>
    <xs:attribute name="priority" type="xs:nonNegativeInteger"/>
    <xs:attribute name="save_all_attempts" type="xs:boolean"/>
    <xs:attribute name="time_interval" type="xs:string"/>
    <xs:attribute name="min_duration" type="xs:nonNegativeInteger"/>
    <xs:attribute name="group" type="xs:string"/>
    <xs:attribute name="shuffle" type="xs:boolean"/>
  </xs:complexType>

```



```
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="template" type="templateType"/>
  </xs:all>
</xs:complexType>

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="result" type="resultType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>
```

Запрос:

http://192.168.1.21:9999/commands/dialer_template_info

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_dialer_template_info.xsd">
  <request domain="biysk.local">
    <template id="063f86fccbeae10b"/>
  </request>
</in>
```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_dialer_template_info.xsd">
  <result>
    <template id="063f86fccbeae10b" name="info4lab" description="info for labs"
      script="play3" script_id="0631ad22420a1493" call_success_by_variable="" shuffle="false"
      lines_limit="0" calls_limit="1" recall_timeout="900" no_answer_timeout="20" priority="4"
      save_all_attempts="true" min_duration="15" time_interval="09:00-18:00">
      <numbers>
        <number digits="240464"/>
        <number digits="416977"/>
      </numbers>
      <results>
        <script id="0631ad22420a1493">
          <block name="begin_1" output="true" result=""/>
          <block name="play_2" output="true" result=""/>
        </script>
      </results>
      <operator_numbers>
        <number digits="240101"/>
      </operator_numbers>
      <property_descriptions>
        <property name="number_digits_meta" data_name="Номер" report_name=""/>
      </property_descriptions>
    </template>
  </result>
</out>

```

hc_dialer_template_set

Команда предназначена для изменения свойств шаблона дозвона.

Аналог команды в CoCon:

`/domain/<DOMAIN>/ss/dialer/template/set`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/dialer_template_set

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404, 434 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="dialer_common.xsd"/>
  <!-- campaignType -->
  <xs:complexType name="templateType">
    <xs:all>
      <xs:element name="numbers" type="numbersType" minOccurs="0"/>
      <xs:element name="results" type="scriptsType" minOccurs="0"/>
      <xs:element name="operator_numbers" type="operatorNumbersType" minOccurs="0"/>
      <xs:element name="property_descriptions" type="propertyDescriptionsType"
minOccurs="0"/>
    </xs:all>
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string"/>
    <xs:attribute name="description" type="xs:string"/>
    <xs:attribute name="script_id" type="xs:string"/>
    <xs:attribute name="call_success_by_variable" type="xs:string"/>
    <xs:attribute name="operator_number" type="xs:string"/>
    <xs:attribute name="lines_limit" type="xs:nonNegativeInteger"/>
    <xs:attribute name="calls_limit" type="xs:nonNegativeInteger"/>
    <xs:attribute name="recall_timeout" type="xs:nonNegativeInteger"/>
    <xs:attribute name="no_answer_timeout" type="xs:nonNegativeInteger"/>
    <xs:attribute name="priority" type="xs:nonNegativeInteger"/>
    <xs:attribute name="save_all_attempts" type="xs:boolean"/>
    <xs:attribute name="time_interval" type="xs:string"/>
    <xs:attribute name="min_duration" type="xs:nonNegativeInteger"/>
    <xs:attribute name="group" type="xs:string"/>
    <xs:attribute name="shuffle" type="xs:boolean"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="template" type="templateType"/>
            </xs:all>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:complexType name="responseType">
    <xs:attribute name="result" type="xs:string" use="required"/>

```

```

</xs:complexType>

<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="response" type="responseType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/dialer_template_set

```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_dialer_template_set.xsd">
  <request domain="biysk.local">
    <template id="063f86fccbeae10b" name="info4lab" description="info for labs"
  script_id="0631ad22420a1493">
      <numbers>
        <number digits="240464">
          <property name="debt" value="500"/>
          <property name="name" value="Светлана"/>
        </number>
        <number digits="416977">
          <property name="debt" value="500"/>
          <property name="name" value="Александр"/>
        </number>
      </numbers>
      <property_descriptions>
        <property name="debt" data_name="debts" report_name="Задолженность"/>
        <property name="name" data_name="FIO"/>
      </property_descriptions>
    </template>
  </request>
</in>

```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_dialer_template_set.xsd">
  <response result="ok"/>
</out>
```

[Hc_dialer_template_list](#)

Команда предназначена для просмотра списка шаблонов дозвона.

Аналог команды в CoCon:

`/domain/<DOMAIN>/ss/dialer/template/list`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/dialer_template_list

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="dialer_common.xsd"/>
  <xs:complexType name="templateType">
    <xs:all>
      <xs:element name="operator_numbers" type="operatorNumbersType" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
    <xs:attribute name="lines_limit" type="xs:string"/>
    <xs:attribute name="calls_limit" type="xs:string"/>
    <xs:attribute name="recall_timeout" type="xs:string"/>
    <xs:attribute name="priority" type="xs:string"/>
    <xs:attribute name="call_success_by_variable" type="xs:string"/>
    <xs:attribute name="time_interval" type="xs:string"/>
    <xs:attribute name="script" type="xs:string"/>
  </xs:complexType>
  <xs:complexType name="templatesType">
    <xs:sequence>
      <xs:element name="template" type="templateType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="format" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="result" type="templatesType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/dialer_template_list

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_dialer_template_list.xsd">
  <request domain="biysk.local" format="full"/>
</in>
```

ОТВЕТ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_dialer_template_list.xsd">
  <result>
    <template id="062d88f9c13c91d4" name="test2" description="jhgk" script="market_dialer"
call_success_by_variable="" lines_limit="1" calls_limit="1" recall_timeout="1" priority="4"
time_interval="09:00-18:00">
      <operator_numbers>
        <number digits="240244"/>
      </operator_numbers>
    </template>
    <template id="0627ec911ac5a8ee" name="опрос" description="реклама для абонентов"
script="obzvон" call_success_by_variable="10" lines_limit="1" calls_limit="2"
recall_timeout="900" priority="4" time_interval="09:00-11:00">
      <operator_numbers>
        <number digits="240469"/>
      </operator_numbers>
    </template>
    <template id="062d8904806b398a" name="ustp-1" description="Сбор бригады 1" script="NOT
FOUND (Id = 06202b326f2fd9e8)" call_success_by_variable="" lines_limit="2" calls_limit="2"
recall_timeout="60" priority="2" time_interval="00:00-23:59">
      <operator_numbers>
        <number digits="240101"/>
      </operator_numbers>
    </template>
    <template id="062d87eafcc580cd" name="market1" description="акция по новым услугам"
script="obzvон" call_success_by_variable="" lines_limit="2" calls_limit="1" recall_timeout="60"
priority="4" time_interval="08:00-17:00">
      <operator_numbers>
        <number digits="240244"/>
      </operator_numbers>
    </template>
  </result>
</out>
```

[Hc_dialer_template_remove](#)

Команда предназначена для удаления шаблона дозвона.

Аналог команды в CoCon:

`/domain/<DOMAIN>/ss/dialer/template/remove`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/dialer_template_remove

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- campaignType -->
  <xs:complexType name="templateType">
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="template" type="templateType"/>
            </xs:all>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:complexType name="responseType">
    <xs:attribute name="result" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/dialer_template_remove

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_dialer_template_remove.xsd">
  <request domain="biysk.local">
    <template id="062d88f9c13c91d4"/>
  </request>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_dialer_template_remove.xsd">
  <response result="ok"/>
</out>
```

Hc_direction - Управление направлениями

- [Hc_direction_list](#)
- [Hc_direction_declare](#)
- [Hc_direction_remove](#)
- [Hc_direction_set](#)
- [Hc_direction_info](#)
- [Hc_direction_reset_stat](#)
- [Hc_direction_clean](#)

[Hc_direction_list](#)

Команда позволяет получить список направлений.

Аналог команды в CoCon:

domain/<DOMAIN>/direction/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/direction_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="directionType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="is_active" type="xs:boolean" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="bandwidth" type="xs:integer" use="optional"/>
    <xs:attribute name="trunks" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="domainOutType">
    <xs:sequence>
      <xs:element name="direction" type="directionType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain" type="domainOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

    </xs:element>
  </xs:all>
</xs:complexType>
</xs:element>

</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/direction_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_direction_list.xsd">
<request>
  <domain name="biysk.local"/>
</request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_direction_list.xsd">
  <response>
    <domain name="biysk.local">
      <direction description="ems_trunks" name="ems1" trunks="ems1;ems2" bandwidth="20"
is_active="true"/>
      <direction description="test description" name="dir2" trunks="" bandwidth="10"
is_active="true"/>
      <direction description="to 192.68.2.12" name="bsk" trunks="bsk1;bsk2" bandwidth="50"
is_active="false"/>
    </domain>
  </response>
</out>

```

[Hc_direction_declare](#)

Команда позволяет создать новое направление.

Аналог команды в CoCon:

domain/<DOMAIN>/direction/

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/direction_declare

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>
  <!-- <xs:include schemaLocation="alias_profile_common.xsd"/> -->

  <xs:complexType name="directionType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="is_active" type="xs:boolean" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="bandwidth" type="xs:integer" use="optional"/>
  </xs:complexType>

  <xs:complexType name="domainType">
    <xs:sequence>
      <xs:element name="direction" type="directionType" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!--Positive case-->
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <!--Or error-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/direction_declare

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_direction_declare.xsd">
  <request>
    <domain name="biysk.local">
      <direction name="dir2" description="test description" is_active="true" bandwidth="10"
    />
    </domain>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_direction_declare.xsd">
  <response>
    <ok/>
  </response>
</out>

```

Код ответа:

404

Ответ:

```

<?xml version="1.0" encoding="UTF-8"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_direction_declare.xsd">
  <error cmd="hc_direction_declare" reason="already_declared" body="{error,already_declared}"/>
</out>

```

Нс_direction_remove

Команда позволяет удалить имеющееся направление.

Аналог команды в CoCon:

`domain/<DOMAIN>/direction/remove`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/direction_remove

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
              <xs:element name="direction" type="domainType" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!-- Positive case -->
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <!-- Or error -->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/direction_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_direction_remove.xsd">
<request>
  <domain name="biysk.local" />
  <direction name="dir2" />
</request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_direction_remove.xsd">
  <response>
    <ok/>
  </response>
</out>
```

Код ответа:

404

Ответ:

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_direction_remove.xsd">
  <error cmd="hc_direction_remove" reason="not_exists" body="{error,direction_not_exists}"
  entity="biysk.local"/>
</out>
```

hc_direction_set

Команда позволяет изменить значение свойств направления.

Аналог команды в CoCon:

domain/<DOMAIN>/direction/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/direction_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="trunksType">
    <xs:attribute name="owner" type="xs:string" use="required"/>
    <xs:attribute name="group" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="directionType">
    <xs:choice>
      <xs:element name="trunks" type="trunksType"/>
      <xs:element name="property" type="propertyType"/>
    </xs:choice>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!--
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="is_active" type="xs:boolean" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="bandwidth" type="xs:integer" use="optional"/>
  -->

  <xs:complexType name="domainType">
    <xs:sequence>
      <xs:element name="direction" type="directionType" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

    </xs:complexType>
</xs:element>

<!-- out-->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <!--Positive case-->
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="ok"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <!--0r error-->
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/direction_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_direction_set.xsd">
  <request>
    <domain name="biysk.local">
      <direction name="bsk">
        <trunks owner="*" group="ssw.gr" value="bsk1;bsk2;ems1;ems2"/>
      </direction>
    </domain>
  </request>
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_direction_set.xsd">
  <response>
    <ok/>
  </response>
</out>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_direction_set.xsd">
<request>
  <domain name="biysk.local">
    <direction name="bsk">
      <property name="is_active" value="false"/>
    </direction>
  </domain>
</request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_direction_set.xsd">
  <response>
    <ok/>
  </response>
</out>
```

Код ответа:

404

Ответ:

```
<?xml version="1.0" encoding="UTF-8"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_direction_set.xsd">
  <error cmd="hc_direction_set" reason="badargs" body="{error,unknown_property}"/></out>,
</out>
```

Нс_direction_info

Команда позволяет просмотреть список свойств направления.

Аналог команды в CoCon:

`domain/<DOMAIN>/direction/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/direction_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="domainType">
    <xs:sequence>
      <xs:element name="direction" type="directionType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="directionType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="is_active" type="xs:boolean" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="bandwidth" type="xs:integer" use="optional"/>
    <xs:attribute name="trunks" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="directionOutType">
    <xs:sequence>
      <xs:element name="trunks" type="trunksType" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="is_active" type="xs:boolean" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="bandwidth" type="xs:integer" use="optional"/>
  </xs:complexType>

  <xs:complexType name="domainOutType">
    <xs:sequence>
      <xs:element name="direction" type="directionOutType" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="trunksType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
              <xs:element name="direction" type="domainType" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```



```

        </xs:sequence>
    </xs:complexType>
</xs:element>
    </xs:all>
</xs:complexType>
</xs:element>

<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:all>
            <xs:element name="response">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain" type="domainOutType"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:all>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/direction_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_direction_info.xsd">
  <request>
    <domain name="biysk.local"/>
    <direction name="bsk"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_direction_info.xsd">
  <response>
    <domain name="biysk.local">
      <direction description="to 192.68.2.12" name="bsk" bandwidth="50" is_active="false">
        <trunks value="bsk1;bsk2"/>
      </direction>
    </domain>
  </response>
</out>
```

[Hc_direction_reset_stat](#)

Команда позволяет сбросить статистику по направлению.

Аналог команды в CoCon:

domain/<DOMAIN>/direction/reset-statistics

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/direction_reset_stat

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="optionType">
    <xs:attribute name="name" type="optionProp" use="required"/>
  </xs:complexType>

  <xs:simpleType name="optionProp">
    <xs:restriction base="xs:token">
      <xs:enumeration value="description"/>
      <xs:enumeration value="trunks"/>
      <xs:enumeration value="is_active"/>
      <xs:enumeration value="bandwidth"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
              <xs:element name="direction" type="domainType" minOccurs="1" maxOccurs="1"/>
              <xs:element name="option" type="optionType" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!--Positive case-->
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

        </xs:complexType>
    </xs:element>
    <!--Or error-->
    <xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/direction_reset_stat

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_direction_reset_stat.xsd">
<request>
  <domain name="biysk.local"/>
  <direction name="bsk"/>
  <option name="out"/>
</request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_direction_reset_stat.xsd">
  <response>
    <ok/>
  </response>
</out>

```

Код ответа:

404

Ответ:

```
<?xml version="1.0" encoding="UTF-8"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_direction_reset_stat.xsd">
  <error cmd="hc_direction_reset_stat" reason="not_exists" body="{error,direction_not_exists}"
  entity="d.city"/>
</out>
```

hc_direction_clean

Команда позволяет удалить определенное свойство из направления.

Аналог команды в CoCon:

`domain/<DOMAIN>/direction/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/direction_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="optionType">
    <xs:attribute name="name" type="optionProp" use="required"/>
  </xs:complexType>

  <xs:simpleType name="optionProp">
    <xs:restriction base="xs:token">
      <xs:enumeration value="description"/>
      <xs:enumeration value="trunks"/>
      <xs:enumeration value="is_active"/>
      <xs:enumeration value="bandwidth"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
              <xs:element name="direction" type="domainType" minOccurs="1" maxOccurs="1"/>
              <xs:element name="option" type="optionType" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!--Positive case-->
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

        </xs:complexType>
    </xs:element>
    <!--Or error-->
    <xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/direction_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_direction_clean.xsd">
<request>
  <domain name="biysk.local"/>
  <direction name="bsk"/>
  <option name="is_active"/>
</request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_direction_clean.xsd">
  <response>
    <ok/>
  </response>
</out>

```

При несуществующем направлении

Код ответа:

200

Ответ:

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_direction_clean.xsd">
  <error cmd="hc_direction_clean" reason="not_exists" body="{error,direction_not_exists}"
    entity="biysk.local"/>
</out>
```

Hc_distinctive_ring_rule - Команды управления услугой distinctive_ring

- [Hc_distinctive_ring_rule_info](#)
- [Hc_distinctive_ring_rule_remove](#)
- [Hc_distinctive_ring_rule_set](#)

[Hc_distinctive_ring_rule_info](#)

Команда позволяет получить информацию по созданным правилам для услуги distinctive ring.

Аналог команды в CoCon:

domain/<DOMAIN>/distinctive-ring/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/distinctive_ring_rule_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required" />
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="rule" minOccurs="0" maxOccurs="unbounded">
          <xs:complexType>
            <xs:attribute name="mask" type="xs:string" use="required"/>
            <xs:attribute name="ring" type="xs:string"/>
            <xs:attribute name="path" type="xs:string"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/distinctive_ring_rule_info

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_disti
nctive_ring_rule_info.xsd">
  <request domain="biysk.local"/>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_distinctive_ring_rule_info.xsd">
  <rule mask="basiccall" ring="ring01" path=""/>
  <rule mask="intercitycall" ring="ring02" path="system://sounds/ring/dash.wav"/>
  <rule mask="vipcall" ring="ring09" path="system://sounds/ring/beep.wav,system://sounds/ring/zdravstujte.wav"/>
  <rule mask="3852" ring="ring05" path="system://sounds/ring/nvh.wav"/>
  <rule mask="991" ring="ring13" path="system://sounds/ring/ustrica6.wav"/>
</out>
```

[Hc_distinctive_ring_rule_remove](#)

Команда позволяет удалить правило для услуги distinctive ring.

Аналог команды в CoCon:

domain/<DOMAIN>/distinctive-ring/rule-remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/distinctive_ring_rule_remove

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="rule" minOccurs="1" maxOccurs="16">
                <xs:complexType>
                  <xs:attribute name="mask" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required" />
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/distinctive_ring_rule_remove

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_distinctive_ring_rule_remove.xsd">
  <request domain="biysk.local">
    <rule mask="991"/>
  </request>
</in>

```

Ответ: 200

[Hc_distinctive_ring_rule_set](#)

Команда позволяет изменить правило для услуги distinctive ring.

Аналог команды в CoCon:

domain/<DOMAIN>/distinctive-ring/rule-change

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/distinctive_ring_rule_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="rule" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="mask" type="xs:string" use="required"/>
                  <xs:attribute name="ring" type="ringType"/>
                  <xs:attribute name="path" type="xs:string"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required" />
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:simpleType name="ringType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="ring01"/>
      <xs:enumeration value="ring02"/>
      <xs:enumeration value="ring03"/>
      <xs:enumeration value="ring04"/>
      <xs:enumeration value="ring05"/>
      <xs:enumeration value="ring06"/>
      <xs:enumeration value="ring07"/>
      <xs:enumeration value="ring08"/>
      <xs:enumeration value="ring09"/>
      <xs:enumeration value="ring10"/>
      <xs:enumeration value="ring11"/>
      <xs:enumeration value="ring12"/>
      <xs:enumeration value="ring13"/>
      <xs:enumeration value="ring14"/>
      <xs:enumeration value="ring15"/>
      <xs:enumeration value="ring16"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- out -->
  <xs:element name="out">

```

```

<xs:complexType>
  <xs:sequence>
    <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/distinctive_ring_rule_set

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_disti
nctive_ring_rule_set.xsd">
  <request domain="biysk.local">
    <rule mask="vipcall" ring="ring09" path="system://sounds/ring/beep.wav,system://sounds/
ring/zdravstujte.wav"/>
  </request>
</in>

```

Ответ: 200

Нс_domain_access_type - Команды управления типами доступа

- [Нс_domain_access_type_info](#)
- [Нс_domain_access_type_remove](#)
- [Нс_domain_access_type_declare](#)

[Нс_domain_access_type_info](#)

Команда позволяет просмотреть профили Access type, декларированные в системе.

Аналог команды в CoCon:

/domain/<DOMAIN>/access-type/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_access_type_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="typeInType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="typeType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="private_in" type="xs:boolean" use="required"/>
    <xs:attribute name="private_out" type="xs:boolean" use="required"/>
    <xs:attribute name="local_in" type="xs:boolean" use="required"/>
    <xs:attribute name="local_out" type="xs:boolean" use="required"/>
    <xs:attribute name="zone_in" type="xs:boolean" use="required"/>
    <xs:attribute name="zone_out" type="xs:boolean" use="required"/>
    <xs:attribute name="intercity_in" type="xs:boolean" use="required"/>
    <xs:attribute name="intercity_out" type="xs:boolean" use="required"/>
    <xs:attribute name="international_in" type="xs:boolean" use="required"/>
    <xs:attribute name="international_out" type="xs:boolean" use="required"/>
    <xs:attribute name="emergency_in" type="xs:boolean" use="required"/>
    <xs:attribute name="emergency_out" type="xs:boolean" use="required"/>
    <xs:attribute name="cause" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="type" type="typeInType" minOccurs="1" />
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="type" type="typeType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/domain_access_type_info

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_access_type_info.xsd">
  <type domain="biysk.local" />
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_access_type_info.xsd">
  <type domain="biysk.local" name="admin1" private_in="false" private_out="false"
local_in="false" local_out="false" zone_in="false" zone_out="false" intercity_in="false"
intercity_out="false" international_in="false" international_out="false" emergency_in="true"
emergency_out="true" cause="21" description="admin"/>
  <type domain="biysk.local" name="access_type2" private_in="true" private_out="true"
local_in="true" local_out="true" zone_in="true" zone_out="true" intercity_in="true"
intercity_out="false" international_in="true" international_out="false" emergency_in="true"
emergency_out="true" cause="21" description="Call Rejected"/>
  <type domain="biysk.local" name="access_type1" private_in="true" private_out="true"
local_in="true" local_out="true" zone_in="true" zone_out="true" intercity_in="true"
intercity_out="true" international_in="true" international_out="false" emergency_in="true"
emergency_out="true" cause="21" description="Call Rejected"/>
  <type domain="biysk.local" name="access_type0" private_in="true" private_out="true"
local_in="true" local_out="true" zone_in="true" zone_out="true" intercity_in="true"
intercity_out="true" international_in="true" international_out="true" emergency_in="true"
emergency_out="true" cause="16" description="Normal"/>
</out>
```

Нс_domain_access_type_remove

Команда позволяет удалить профиль Access type.

Аналог команды в CoCon:

/domain/test.domain/access-type/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_access_type_remove

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="typeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="type" type="typeType" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_access_type_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_access_type_remove.xsd">
  <type domain="biysk.local" name="access_type2" />
</in>
```

Ответ: 204

[Hc_domain_access_type_declare](#)

Команда позволяет создать новый профиль Access type.

Аналог команды в CoCon:

`domain/<DOMAIN>/access-type/declare`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_access_type_declare

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить двумя сообщениями:

- 204 – в случае успеха;

- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="typeType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="private_in" type="xs:boolean" use="required"/>
    <xs:attribute name="private_out" type="xs:boolean" use="required"/>
    <xs:attribute name="local_in" type="xs:boolean" use="required"/>
    <xs:attribute name="local_out" type="xs:boolean" use="required"/>
    <xs:attribute name="zone_in" type="xs:boolean" use="required"/>
    <xs:attribute name="zone_out" type="xs:boolean" use="required"/>
    <xs:attribute name="intercity_in" type="xs:boolean" use="required"/>
    <xs:attribute name="intercity_out" type="xs:boolean" use="required"/>
    <xs:attribute name="international_in" type="xs:boolean" use="required"/>
    <xs:attribute name="international_out" type="xs:boolean" use="required"/>
    <xs:attribute name="emergency_in" type="xs:boolean" use="required"/>
    <xs:attribute name="emergency_out" type="xs:boolean" use="required"/>
    <xs:attribute name="cause" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="type" type="typeType" minOccurs="1" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос:

http://192.168.1.21:9999/commands/domain_access_type_declare

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="../priv/
xsd/domain_access_type_declare.xsd">
  <type domain="ct.office" name="first_access" private_in="false" private_out="false"
local_in="false" local_out="false" zone_in="false" zone_out="false"
  intercity_in="false" intercity_out="false" international_in="false"
international_out="false" emergency_in="true" emergency_out="true" cause="16"/>
</in>
```

Ответ: 204

hc_domain_calendar - Управление календарем на уровне домена

- [hc_domain_calendar_set](#)
- [hc_domain_calendar_show](#)

hc_domain_calendar_set

Команда позволяет изменять настройки календаря на уровне домена.

Аналог команды в CoCon:

```
/domain/<DOMAIN>/calendar/set
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_calendar_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- maskType-->
  <xs:complexType name="maskType">
    <xs:attribute name="year" type="xs:string"/>
    <xs:attribute name="month" type="xs:string"/>
    <xs:attribute name="day" type="xs:string"/>
    <xs:attribute name="day_of_week" type="xs:string"/>
    <xs:attribute name="day_of_year" type="xs:string"/>
  </xs:complexType>
  <!-- defaultType-->
  <xs:complexType name="defaultType">
    <xs:all>
      <xs:element name="mask" type="maskType" minOccurs="0"/>
      <xs:element name="day" type="maskdayType" minOccurs="0"/>
    </xs:all>
  </xs:complexType>
  <!-- defaultsType-->
  <xs:complexType name="defaultsType">
    <xs:sequence>
      <xs:element name="default" type="defaultType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string"/>
    <xs:attribute name="value" type="xs:string"/>
  </xs:complexType>
  <!-- propertiesType-->
  <xs:complexType name="propertiesType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- typeType-->
  <xs:simpleType name="typeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="HALF-WORK"/>
      <xs:enumeration value="WORK"/>
      <xs:enumeration value="DAY-OFF"/>
      <xs:enumeration value="HALF-HOLIDAY"/>
      <xs:enumeration value="HOLIDAY"/>
      <xs:enumeration value="HALF-WORK-AND-HOLIDAY"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- maskdayType-->
  <xs:complexType name="maskdayType">
    <xs:all>

```

```

        <xs:element name="properties" type="propertiesType" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="date" type="xs:string"/>
    <xs:attribute name="type" type="typeType"/>
</xs:complexType>
<!-- dayType -->
<xs:complexType name="dayType">
    <xs:all>
        <xs:element name="properties" type="propertiesType" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="year" type="xs:string"/>
    <xs:attribute name="month" type="xs:string"/>
    <xs:attribute name="day" type="xs:string"/>
    <xs:attribute name="type" type="typeType"/>
</xs:complexType>
<!-- daysType -->
<xs:complexType name="daysType">
    <xs:sequence>
        <xs:element name="day" type="dayType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<!-- yearType -->
<xs:complexType name="yearType">
    <xs:all>
        <xs:element name="days" type="daysType" minOccurs="0"/>
        <xs:element name="defaults" type="defaultsType" minOccurs="0"/>
        <xs:element name="properties" type="propertiesType" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="year" type="xs:string"/>
</xs:complexType>
<!-- yearsType -->
<xs:complexType name="yearsType">
    <xs:sequence>
        <xs:element name="year" type="yearType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="calendarType">
    <xs:all>
        <xs:element name="years" type="yearsType" minOccurs="0"/>
        <xs:element name="defaults" type="defaultsType" minOccurs="0"/>
        <xs:element name="properties" type="propertiesType" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="timezone" type="xs:string"/>
</xs:complexType>
<!-- in -->
<xs:element name="in">
    <xs:complexType>
        <xs:all>
            <xs:element name="request">
                <xs:complexType>

```

```
        <xs:all>
            <xs:element name="calendar" type="calendarType"/>
        </xs:all>
        <xs:attribute name="domain" type="xs:string" use="optional"/>
    </xs:complexType>
</xs:element>
</xs:all>
</xs:complexType>
</xs:element>
<!-- out-->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <!--Positive case-->
            <xs:element name="response">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="ok"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>
```

Пример

Запрос: http://192.168.1.21:9999/commands/domain_calendar_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_domain_calendar_set.xsd">
  <request domain="biysk.local">
    <calendar timezone="UTC+00:00">
      <years>
        <year year="2019">
          <days>
            <day year="2019" month="3" day="8" type="HOLIDAY">
              <properties/>
            </day>
            <day year="2019" month="1" day="2" type="HOLIDAY">
              <properties/>
            </day>
          </days>
          <defaults/>
          <properties/>
        </year>
        <year year="2020">
          <days>
            <day year="2020" month="9" day="18" type="HALF-WORK">
              <properties/>
            </day>
            <day year="2020" month="3" day="7" type="HALF-HOLIDAY">
              <properties/>
            </day>
          </days>
          <defaults/>
          <properties/>
        </year>
      </years>
      <defaults>
        <default>
          <mask year="" month="5" day="9" day_of_week="" day_of_year=""/>
          <day date="" type="HOLIDAY">
            <properties/>
          </day>
        </default>
        <default>
          <mask year="" month="1" day="" day_of_week="" day_of_year="8 9 10 11 12"/>
          <day date="" type="WORK">
            <properties/>
          </day>
        </default>
        <default>
          <mask year="" month="" day="" day_of_week="6 7" day_of_year=""/>
          <day date="" type="DAY-OFF">
            <properties/>
          </day>
        </default>
      </defaults>
    </calendar>
  </request>
</in>

```

```
        </day>
      </default>
    </defaults>
    <properties>
      <property key="description" value="Default system calendar"/>
    </properties>
  </calendar>
</request>
</in>
```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_domain_calendar_set.xsd">
  <response>
    <ok/>
  </response>
</out>
```

[Hc_domain_calendar_show](#)

Команда для просмотра календаря домена.

Аналог команды в CoCon:

`/domain/<DOMAIN>/calendar/show`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_calendar_show

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- maskType -->
  <xs:complexType name="maskType">
    <xs:attribute name="year" type="xs:string"/>
    <xs:attribute name="month" type="xs:string"/>
    <xs:attribute name="day" type="xs:string"/>
    <xs:attribute name="day_of_week" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Space delimited list of days. 1 - monday, 7 - sunday
          Example: 1 2 3 4 5
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="day_of_year" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Space delimited list of days.
          Example: 256 257
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- defaultType -->
  <xs:complexType name="defaultType">
    <xs:all>
      <xs:element name="mask" type="maskType" minOccurs="0"/>
      <xs:element name="day" type="maskdayType" minOccurs="0"/>
    </xs:all>
  </xs:complexType>
  <!-- defaultsType -->
  <xs:complexType name="defaultsType">
    <xs:sequence>
      <xs:element name="default" type="defaultType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- propertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string"/>
    <xs:attribute name="value" type="xs:string"/>
  </xs:complexType>
  <!-- propertiesType -->
  <xs:complexType name="propertiesType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

```

```

<!-- typeType-->
<xs:simpleType name="typeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="HALF-WORK"/>
    <xs:enumeration value="WORK"/>
    <xs:enumeration value="DAY-OFF"/>
    <xs:enumeration value="HALF-HOLIDAY"/>
    <xs:enumeration value="HOLIDAY"/>
    <xs:enumeration value="HALF-WORK-AND-HOLIDAY"/>
  </xs:restriction>
</xs:simpleType>
<!-- maskdayType-->
<xs:complexType name="maskdayType">
  <xs:all>
    <xs:element name="properties" type="propertiesType" minOccurs="0"/>
  </xs:all>
  <xs:attribute name="date" type="xs:string"/>
  <xs:attribute name="type" type="typeType"/>
</xs:complexType>
<!-- dayType-->
<xs:complexType name="dayType">
  <xs:all>
    <xs:element name="properties" type="propertiesType" minOccurs="0"/>
  </xs:all>
  <xs:attribute name="year" type="xs:string"/>
  <xs:attribute name="month" type="xs:string"/>
  <xs:attribute name="day" type="xs:string"/>
  <xs:attribute name="type" type="typeType"/>
</xs:complexType>
<!-- daysType-->
<xs:complexType name="daysType">
  <xs:sequence>
    <xs:element name="day" type="dayType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!-- yearType-->
<xs:complexType name="yearType">
  <xs:all>
    <xs:element name="days" type="daysType" minOccurs="0"/>
    <xs:element name="defaults" type="defaultsType" minOccurs="0"/>
    <xs:element name="properties" type="propertiesType" minOccurs="0"/>
  </xs:all>
  <xs:attribute name="year" type="xs:string"/>
</xs:complexType>
<!-- yearsType-->
<xs:complexType name="yearsType">
  <xs:sequence>
    <xs:element name="year" type="yearType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

```

```

<xs:complexType name="calendarType">
  <xs:all>
    <xs:element name="years" type="yearsType" minOccurs="0"/>
    <xs:element name="defaults" type="defaultsType" minOccurs="0"/>
    <xs:element name="properties" type="propertiesType" minOccurs="0"/>
  </xs:all>
  <xs:attribute name="timezone" type="xs:string"/>
</xs:complexType>
<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:attribute name="domain" type="xs:string" use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <!--Positive case-->
      <xs:element name="calendar" type="calendarType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/domain_calendar_show

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_domain_calendar_show.xsd">
  <request domain="biysk.local"/>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_domain_calendar_show.xsd">
  <calendar timezone="UTC+00:00">
    <years>
      <year year="2019">
        <days>
          <day year="2019" month="1" day="2" type="HOLIDAY">
            <properties/>
          </day>
          <day year="2019" month="3" day="8" type="HOLIDAY">
            <properties/>
          </day>
        </days>
        <defaults/>
        <properties/>
      </year>
      <year year="2020">
        <days>
          <day year="2022" month="3" day="7" type="HALF-HOLIDAY">
            <properties/>
          </day>
          <day year="2022" month="9" day="18" type="HALF-WORK">
            <properties/>
          </day>
        </days>
        <defaults/>
        <properties/>
      </year>
    </years>
    <defaults>
      <default>
        <mask year="*" month="*" day="*" day_of_week="6 7" day_of_year="*"/>
        <day date="*" type="DAY-OFF">
          <properties/>
        </day>
      </default>
      <default>
        <mask year="*" month="5" day="9" day_of_week="*" day_of_year="*"/>
        <day date="*" type="HOLIDAY">
          <properties/>
        </day>
      </default>
      <default>
        <mask year="*" month="1" day="*" day_of_week="*" day_of_year="8 9 10 11 12"/>
        <day date="*" type="WORK">
          <properties/>
        </day>
    </defaults>
  </calendar>
</out>

```

```
        </default>
    </defaults>
    <properties>
        <property key="description" value="Default system calendar"/>
    </properties>
</calendar>
</out>
```

Hc_domain_call_statistics - Команда для получения значения общей статистики вызовов домена

Hc_domain_call_statistics

Команда для получения значения общей статистики вызовов домена.

Аналог команды в CoCon:

domain/<DOMAIN>/licence/usage

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_call_statistics

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>
  <!-- idType -->
  <xs:simpleType name="idType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="cp">
        <xs:annotation>
          <xs:documentation xml:lang="en">
            Simple calls
          </xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="tc">
        <xs:annotation>
          <xs:documentation xml:lang="en">
            Teleconference calls
          </xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="cc-active-agents">
        <xs:annotation>
          <xs:documentation xml:lang="en">
            Logged in call center operators
          </xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="cc-active-supervisors">
        <xs:annotation>
          <xs:documentation xml:lang="en">
            Logged in call center supervisors
          </xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="tsmn-concurrent-calls">
        <xs:annotation>
          <xs:documentation xml:lang="en">
            Call on TSMN service
          </xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="tsmn-concurrent-calls-redundancy">
        <xs:annotation>
          <xs:documentation xml:lang="en">
            Call on TSMN service
          </xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="add-on-conferences">

```

```

    <xs:annotation>
      <xs:documentation xml:lang="en">
        Add-On conferences
      </xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="tc-active-conferences">
    <xs:annotation>
      <xs:documentation xml:lang="en">
        Teleconferences
      </xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="meet-me-conferences">
    <xs:annotation>
      <xs:documentation xml:lang="en">
        Add-On conferences
      </xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="chat-room-conferences">
    <xs:annotation>
      <xs:documentation xml:lang="en">
        Chat rooms
      </xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="dialer-channels">
    <xs:annotation>
      <xs:documentation xml:lang="en">
        Calls from dialer subsystem
      </xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="recorder-voice-channels">
    <xs:annotation>
      <xs:documentation xml:lang="en">
        Total count of simultaneous records voice calls
      </xs:documentation>
    </xs:annotation>
  </xs:enumeration>
</xs:restriction>
</xs:simpleType>
<!-- statisticType -->
<xs:complexType name="statisticType">
  <xs:attribute name="id" type="idType" use="required"/>
  <xs:attribute name="activeCalls" type="xs:token" use="optional"/>
  <xs:attribute name="topActiveCalls" type="xs:token" use="optional"/>
  <xs:attribute name="callLimit" type="xs:token" use="optional"/>
  <xs:attribute name="totalCalls" type="xs:token" use="optional"/>

```

```

    <xs:attribute name="rejectedCalls" type="xs:token" use="optional"/>
    <xs:attribute name="cps" type="xs:token" use="optional"/>
</xs:complexType>
<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:attribute name="domain" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <!--Positive case-->
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="statistic" type="statisticType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <!--Or error-->
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/domain_call_statistics

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_domain_
call_statistics.xsd">
  <request domain="biysk.local"/>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_domain_
call_statistics.xsd">
  <response>
    <statistic id="add-on-conferences" rejectedCalls="0" totalCalls="0" cps="0" activeCalls="
0" topActiveCalls="0"/>
    <statistic id="cc-active-agents" rejectedCalls="0" totalCalls="0" cps="0" activeCalls="0"
topActiveCalls="0"/>
    <statistic id="cc-active-supervisors" rejectedCalls="0" totalCalls="0" cps="0"
activeCalls="0" topActiveCalls="0"/>
    <statistic id="chat-room-conferences" rejectedCalls="0" totalCalls="0" cps="0"
activeCalls="0" topActiveCalls="0"/>
    <statistic id="cp" rejectedCalls="0" totalCalls="30" callLimit="100" cps="0" activeCalls="
0" topActiveCalls="2"/>
    <statistic id="dialer-channels" rejectedCalls="0" totalCalls="0" cps="0" activeCalls="0"
topActiveCalls="0"/>
    <statistic id="meet-me-conferences" rejectedCalls="0" totalCalls="0" cps="0" activeCalls="
0" topActiveCalls="0"/>
    <statistic id="recorder-voice-channels" rejectedCalls="0" totalCalls="30" callLimit="200"
cps="0" activeCalls="0" topActiveCalls="1"/>
    <statistic id="tc" rejectedCalls="0" totalCalls="0" cps="0" activeCalls="0"
topActiveCalls="0"/>
    <statistic id="tc-active-conferences" rejectedCalls="0" totalCalls="0" cps="0"
activeCalls="0" topActiveCalls="0"/>
    <statistic id="tsmn-concurrent-calls" rejectedCalls="0" totalCalls="0" cps="0"
activeCalls="0" topActiveCalls="0"/>
    <statistic id="tsmn-concurrent-calls-redundancy" rejectedCalls="0" totalCalls="0" cps="0"
activeCalls="0" topActiveCalls="0"/>
  </response>
</out>

```

Hc_domain_cc_options - Управление настройками ограничений колл-центра домена

- [Hc_domain_cc_options_clean](#)
- [Hc_domain_cc_options_info](#)
- [Hc_domain_cc_options_set](#)

Hc_domain_cc_options_clean

Команда подчистки настроек ограничений Call-центра домена.

Аналог команды в CoCon:

`/domain/<DOMAIN>/cc/properties/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_cc_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemCleanOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/domain_cc_options_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_cc_options_clean.xsd">
  <request>
    <domain_options>
      <system>
        <option name="ldap_server_id" />
      </system>
      <domain name="biysk.local">
        <option name="ldap_server_id" />
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_cc_options_clean.xsd">
  <result>
    <domain_options>
      <system>
        <success option="ldap_server_id"/>
      </system>
      <domain name="biysk.local">
        <success option="ldap_server_id"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_domain_cc_options_info

Команда просмотра настроек ограничений Call-центра домена.

Аналог команды в CoCon:

/domain/<DOMAIN>/cc/properties/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_cc_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show domain properties.
      Usage: http://server:9999/commands/domain\_cc\_options\_info.
      Implementation: hc_domain_cc_options_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- capabilitiesType -->
  <xs:simpleType name="capabilitiesType">
    <xs:restriction base="unionType">
      <xs:enumeration value="all"/>
      <xs:enumeration value="[operator_queue_calls, true]"/>
      <xs:enumeration value="[operator_queue_calls, false]"/>
      <xs:enumeration value="[operator_intervention, true]"/>
      <xs:enumeration value="[operator_intervention, false]"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- domainPropertiesOptionsType -->
  <xs:complexType name="domainCcPropertiesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="capabilities" type="capabilitiesType" use="optional"/>
        <xs:attribute name="cc_pubsub_user" type="stringType" use="optional"/>
        <xs:attribute name="cc_pubsub_password" type="stringType" use="optional"/>
        <xs:attribute name="generate_workitem_id_by_queue" type="booleanType" use="option
al"/>
        <xs:attribute name="ldap_authentication_enabled" type="booleanType" use="optional
"/>
        <xs:attribute name="ldap_server_id" type="stringType" use="optional"/>
        <xs:attribute name="relative_rating_0_1" type="stringType" use="optional"/>
        <xs:attribute name="relative_rating_0_9" type="stringType" use="optional"/>
        <xs:attribute name="relative_rating_1_5" type="stringType" use="optional"/>
        <xs:attribute name="workitem_isup_key" type="binaryType" use="optional"/>
        <xs:annotation>
          <xs:documentation xml:lang="en">
            The list of domain's call center public names splitted by comma.
          </xs:documentation>
        </xs:annotation>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoInType" minOccurs="
1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/domain_cc_options_info

```

<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_domain_cc_options_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_cc_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="domainCcPropertiesOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="domainCcPropertiesOptionsType" workitem_isup_key=""
          relative_rating_1_5="negative: 1 - 2 neutral: 3 - 4 positive: 5" relative_rating_0_9="negative:
0 - 4 neutral: 5 - 6 positive: 7 - 9" relative_rating_0_1="negative: 0 positive: 1"
          ldap_server_id="undefined" ldap_authentication_enabled="false"
          generate_workitem_id_by_queue="false" cc_pubsub_user="pubsub" cc_pubsub_password="pubsub"
          capabilities="all"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_domain_cc_options_set

Команда изменения настроек ограничений Call-центра домена.

Аналог команды в CoCon:

/domain/<DOMAIN>/cc/properties/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_cc_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set domain limits.
      Usage: http://server:9999/commands/domain_cc_options_set.
      Implementation: hc_domain_cc_options_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- domainPropertiesOptionsType -->
  <xs:complexType name="domainCcPropertiesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="generate_workitem_id_by_queue" type="booleanType"
use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              .
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="ldap_authentication_enabled" type="booleanType"
use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              .
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="cc_pubsub_user" type="stringType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              .
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="cc_pubsub_password" type="stringType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              .
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="capabilities" type="stringType" use="optional">

```

```

        <xs:annotation>
            <xs:documentation xml:lang="en">
                .
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="ldap_server_id" type="stringType" use="optional">
        <xs:annotation>
            <xs:documentation xml:lang="en">
                .
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="workitem_isup_key" type="binaryType" use="optional">
        <xs:annotation>
            <xs:documentation xml:lang="en">
                .
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="relative_rating_0_1" type="stringType" use="optional">
        <xs:annotation>
            <xs:documentation xml:lang="en">
                .
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="relative_rating_0_9" type="stringType" use="optional">
        <xs:annotation>
            <xs:documentation xml:lang="en">
                .
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="relative_rating_1_5" type="stringType" use="optional">
        <xs:annotation>
            <xs:documentation xml:lang="en">
                .
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
    <xs:all>
        <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
</xs:complexType>

```

```

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
          <xs:attribute name="force" type="xs:boolean"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/domain_cc_options_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_cc_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="domainCcPropertiesOptionsType" />
      </system>
      <domain name="biysk.local">
        <options xs:type="domainCcPropertiesOptionsType"
ldap_authentication_enabled="false" generate_workitem_id_by_queue="false" cc_pubsub_user="pubsub"
cc_pubsub_password="pubsub" capabilities="all" ldap_server_id="undefined" workitem_isup_key=""
relative_rating_0_1="0 1" relative_rating_0_9="4 7" relative_rating_1_5="2 5"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_cc_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="capabilities"/>
        <success option="cc_pubsub_password"/>
        <success option="cc_pubsub_user"/>
        <success option="generate_workitem_id_by_queue"/>
        <success option="ldap_authentication_enabled"/>
        <success option="ldap_server_id"/>
        <success option="relative_rating_0_1"/>
        <success option="relative_rating_0_9"/>
        <success option="relative_rating_1_5"/>
        <success option="workitem_isup_key"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Hc_domain_data - Управление свойствами домена

- [Hc_domain_data_info](#)
- [Hc_domain_data_set](#)
- [Hc_domain_data_clean](#)

Hc_domain_data_info

Команда для просмотра информации об определенном домене.

Аналог команды в CoCon:

domain/<DOMAIN>/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_data_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;

- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of domain.
      Usage: http://server:9999/commands/domain_data_info.
      Implementation: hc_notifier_options_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="DomainDataType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="pin" type="xs:string" use="optional"/>
        <xs:attribute name="uid" type="xs:string" use="optional"/>
        <xs:attribute name="alias_profile_default" type="xs:string" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainInfoInType" minOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>

```

```

        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1

Запрос: http://192.168.1.21:9999/commands/domain_data_info

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_domain_data_info.xsd">
<request>
    <domain_options domain="biysk.local"/>
</request>
</in>

```

Код ответа: 200

Ответ:

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="hc_domain_data_info.xsd">
    <result>
        <domain_options>
            <system>
                <options xs:type="DomainDataType"/>
            </system>
            <domain name="biysk.local">
                <options xs:type="DomainDataType" alias_profile_default="user_default"/>
            </domain>
        </domain_options>
    </result>
</out>

```

Hc_domain_data_set

Команда предназначена для редактирования свойств домена.

Аналог команды в CoCon:

domain/<DOMAIN>/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_data_set

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set domain properties
      Usage: http://server:9999/commands/domain_data_set.
      Implementation: hc_domain_data_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="DomainDataType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="pin" type="xs:string" use="optional"/>
        <xs:attribute name="alias_profile_default" type="xs:string" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_data_set

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_domain_data_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="DomainDataType"/>
      </system>
      <domain name="test.biysk">
        <options xs:type="DomainDataType" pin="291067"
alias_profile_default="user_default"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Код ответа: 200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_domain_data_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="alias_profile_default"/>
        <success option="pin"/>
      </domain>
    </domain_options>
  </result>
</out>

```

[Hc_domain_data_clean](#)

Команда для сброса в значение по умолчанию доменных свойств.

Аналог команды в CoCon:

```
domain/<DOMAIN>/clean
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_data_clean

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean properties of domain
      Usage: http://server:9999/commands/domain_data_clean.
      Implementation: hc_notifier_options_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/domain_data_clean

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_domain_data_clean.xsd">
<request>
  <domain_options>
    <system />
    <domain name="biysk.local">
      <option name="pin"/>
      <option name="alias_profile_default"/>
    </domain>
  </domain_options>
</request>
</in>
```

Код ответа: 200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_domain_data_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="alias_profile_default"/>
        <success option="pin"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Hc_domain_group - Управление группами доступа

- [Hc_domain_group_list](#) [Hc_domain_group_add](#) [Hc_domain_group_remove](#)
[Hc_domain_group_add_access](#) [Hc_domain_group_remove_access](#)
[Hc_domain_group_access_matrix](#) [Hc_domain_group_list](#)
- [Hc_domain_group_add](#)
- [Hc_domain_group_remove](#)
- [Hc_domain_group_add_access](#)
- [Hc_domain_group_remove_access](#)
- [Hc_domain_group_access_matrix](#)

[Hc_domain_group_list](#)

Команда позволяет просмотреть список групп доступа, задекларированных в данной виртуальной АТС.

Аналог команды в CoCon:

`/domain/<DOMAIN>/access-group/ls-groups`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_group_list

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="groupType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="group" type="groupType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_group_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_group_list.xsd">
  <domain name="biysk.local" />
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_group_list.xsd">
  <group name="all" description="Default access group"/>
</out>

```

Нс_domain_group_add

Команда позволяет создать новую группу доступа в виртуальной АТС.

Аналог команды в CoCon:

`/domain/<DOMAIN>/access-group/add`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_group_add

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="groupType">
    <xs:attribute name="group_a" type="xs:string" use="required"/>
    <xs:attribute name="group_b" type="xs:string" use="required"/>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="group" type="groupType" minOccurs="1" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_group_add


```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_group_add.xsd">
  <group domain="ct.office" name="group3" description="Group3"/>
  <group domain="ct.office" name="group4" description="Group4"/>
</in>
```

Нс_domain_group_remove

Команда позволяет удалить группу доступа из виртуальной АТС.

Аналог команды в CoCon:

/domain/<DOMAIN>/access-group/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_group_remove

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="groupType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="group" type="groupType" minOccurs="1" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.23.34:9999/commands/domain_group_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_group_remove.xsd">
  <group domain="d.city" name="group3" />
</in>
```

Нс_domain_group_add_access

Команда позволяет добавить возможность доступа из группы доступа X в группу доступа Y.

Аналог команды в CoCon:

/domain/<DOMAIN>/access-group/add-access

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_group_add_access

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="groupType">
    <xs:attribute name="group_a" type="xs:string" use="required"/>
    <xs:attribute name="group_b" type="xs:string" use="required"/>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="group" type="groupType" minOccurs="1" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_group_add_access

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_group_add_access.xsd">
  <group domain="biysk.local" group_a = "group1" group_b = "group2" />
</in>
```

Ис domain_group_remove_access

Команда позволяет удалить возможность доступа из группы доступа X в группу доступа Y.

Аналог команды в CoCon:

/domain/<DOMAIN>/access-group/remove-access

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_group_remove_access

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="groupType">
    <xs:attribute name="group_a" type="xs:string" use="required"/>
    <xs:attribute name="group_b" type="xs:string" use="required"/>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="group" type="groupType" minOccurs="1" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_group_remove_access

```
<?xml version='1.0' encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <group group_a="group2" domain="biysk.local" group_b="group3"/>
</in>
```

Нс_domain_group_access_matrix

Команда позволяет просмотреть матрицу доступа группы.

Аналог команды в CoCon:

/domain/<DOMAIN>/access-group/access-matrix

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_group_access_matrix

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="toType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="groupType">
    <xs:sequence>
      <xs:element name="to" type="toType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="group" type="groupType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_group_access_matrix

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_group_access_matrix.xsd">
  <domain name="biysk.local" />
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_group_access_matrix.xsd">
  <group name="all">
    <to name="all"/>
    <to name="group1"/>
    <to name="group2"/>
    <to name="group3"/>
    <to name="group4"/>
  </group>
  <group name="group1">
    <to name="all"/>
    <to name="group2"/>
  </group>
  <group name="group2">
    <to name="all"/>
    <to name="group2"/>
  </group>
  <group name="group3">
    <to name="all"/>
    <to name="group3"/>
    <to name="group4"/>
  </group>
  <group name="group4">
    <to name="all"/>
    <to name="group3"/>
    <to name="group4"/>
  </group>
</out>

```

hc_domain_ldap - Команды управления LDAP/AD сервером

- [hc_domain_ldap_declare](#)
- [hc_domain_ldap_info](#)
- [hc_domain_ldap_remove](#)
- [hc_domain_ldap_set](#)

[hc_domain_ldap_declare](#)

Команда позволяет создавать новую конфигурацию до LDAP/AD сервера.

Аналог команды в CoCon:

/domain/<DOMAIN>/ldap/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_ldap_declare

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <xs:complexType name="ldapOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="id" type="xs:string"/>
        <xs:attribute name="type" type="xs:string"/>
        <xs:attribute name="host" type="xs:string"/>
        <xs:attribute name="port" type="xs:string"/>
        <xs:attribute name="dc_or_domain" type="xs:string"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="responseType">
    <xs:attribute name="result" type="xs:string"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="responseType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:** http://192.168.1.21:9999/commands/domain_ldap_declare


```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_domain_ldap_declare.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="ldapOptionsType"/>
      </system>
      <domain name="test">
        <options dc_or_domain="test" host="ldap.maas.eltex.loc" id="maas.eltex.loc"
port="3899" type="LDAP" DC_OR_DOMAIN ="dc=eltex,dc=loc" ssl="true" xs:type="ldapOptionsType"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_domain_
ldap_declare.xsd">
  <response result="ok"/>
</out>

```

hc_domain_ldap_info

Команда позволяет просматривать новую конфигурации до LDAP/AD сервера.

Аналог команды в CoCon:

/domain/<DOMAIN>/ldap/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_ldap_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="ldapOptionsType">
    <xs:attribute name="id" type="xs:string"/>
    <xs:attribute name="type" type="xs:string"/>
    <xs:attribute name="host" type="xs:string"/>
    <xs:attribute name="domain" type="xs:string"/>
    <xs:attribute name="port" type="xs:string"/>
    <xs:attribute name="use_ssl" type="xs:boolean"/>
    <xs:attribute name="base_search_dn" type="xs:string"/>
    <xs:attribute name="login_attribute" type="xs:string"/>
    <xs:attribute name="recursive_search" type="xs:boolean"/>
    <xs:attribute name="search_auth" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          usage:
            anonymous | Login:Password
            In case of strong security restrictions password may be shown as *****
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="sync_auth" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          usage:
            anonymous | Login:Password
            In case of strong security restrictions password may be shown as *****
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="ssl_certificate" type="xs:string"/>
  </xs:complexType>

  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:sequence minOccurs="0">
      <xs:element name="options" type="ldapOptionsType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>

```

```

        <xs:attribute name="domain" type="xs:string"/>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_ldap_info

```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_domain_
ldap_info.xsd">
  <request domain="test"/>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_domain_
ldap_info.xsd">
  <result>
    <options xs:type="ldapOptionsType" type="ldap" id="1" host="192.168.1.12" port="default"
base_search_dn="dc" login_attribute="uid" recursive_search="true" search_auth="user1:*****"
use_ssl="false" ssl_certificate="undefined"/>
  </result>
</out>

```

hc_domain_ldap_remove

Команда для удаления настроек конфигурации до LDAP/AD сервера

Аналог команды в CoCon:

/domain/<DOMAIN>/ldap/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:http://host:port/commands/domain_ldap_remove**Код HTTP-ответа:**

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="responseType">
    <xs:attribute name="result" type="xs:string"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="connection" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="id" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required" />
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="responseType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:** http://192.168.1.21:9999/commands/domain_ldap_remove

```
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_domain_
  ldap_remove.xsd">
  <request domain="test">
    <connection id="maas.eltex.loc"/>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_domain_
  ldap_remove.xsd">
  <response result="ok"/>
</out>
```

hc_domain_ldap_set

Команда для изменения настроек конфигурации до LDAP/AD сервера

Аналог команды в CoCon:**/domain/<DOMAIN>/ldap/set****Метод HTTP-запроса:**

POST

Шаблон HTTP-запроса:http://host:port/commands/domain_ldap_set**Код HTTP-ответа:**

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="ldapOptionsType">
    <xs:attribute name="host" type="xs:string"/>
    <xs:attribute name="port" type="xs:string"/>
    <xs:attribute name="use_ssl" type="xs:boolean"/>
    <xs:attribute name="base_search_dn" type="xs:string"/>
    <xs:attribute name="login_attribute" type="xs:string"/>
    <xs:attribute name="recursive_search" type="xs:boolean"/>
    <xs:attribute name="domain" type="xs:string"/>
    <xs:attribute name="ssl_certificate" type="xs:string"/>
    <xs:attribute name="search_auth" type="xs:string"/>
    <xs:attribute name="sync_auth" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          search_auth example:
            search_auth="anonymous";
            search_auth="uid=test,ou=System,dc=eltex,dc=loc:test123"
          sync_auth example:
            search_auth="login:password"
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <xs:complexType name="successType">
    <xs:attribute name="id" type="xs:string"/>
    <xs:attribute name="property" type="xs:string"/>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:sequence minOccurs="0" maxOccurs="unbounded">
      <xs:choice>
        <xs:element name="success" type="successType" maxOccurs="unbounded"/>
        <xs:element name="error" type="errorType" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="connectionType">
    <xs:sequence minOccurs="0">
      <xs:element name="options" type="ldapOptionsType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>

```

```

        <xs:element name="request">
            <xs:complexType>
                <xs:sequence maxOccurs="unbounded">
                    <xs:element name="connection" type="connectionType"
maxOccurs="unbounded"/>
                </xs:sequence>
                <xs:attribute name="domain" type="xs:string"/>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:all>
            <xs:element name="result" type="resultType"/>
        </xs:all>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_ldap_set

```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_domain_
ldap_set.xsd">
  <request domain="test">
    <connection id="maas.eltex.loc">
      <options host="ldap.maas.eltex.loc" port="3896" />
    </connection>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_domain_
ldap_set.xsd">
  <result>
    <success id="maas.eltex.loc" property="port"/>
    <success id="maas.eltex.loc" property="host"/>
  </result>
</out>

```

Нс_domain_limits - Управление ограничениями в домене

- [Нс_domain_limits_clean](#)
- [Нс_domain_limits_info](#)
- [Нс_domain_limits_set](#)

Нс_domain_limits_clean

Команда возвращает указанные ограничения, заданные на определенном домене (команда доступна только для администратора ECSS-10), в значения по умолчанию. В текущей версии доступны следующие ограничения:

| Название свойства | Значение по умолчанию | Описание |
|----------------------------------|---------------------------------|---|
| alias_limit | infinity (ограничено лицензией) | Общее количество абонентов (в том числе и виртуальных) в данной виртуальной АТС. |
| call_limit | infinity (ограничено лицензией) | Общее количество одновременно активных вызовов для данной виртуальной АТС. |
| virtual_alias_limit | infinity (ограничено лицензией) | Общее количество виртуальных абонентов в данной виртуальной АТС. |
| digitmap | | Список масок набора, по которому будет валидироваться алиасы при создании. Описание параметра приведено на странице / domain/ – команды управления виртуальными АТС |
| failover | true | Необходимость в резервировании вызовов на данной виртуальной АТС. Параметр используется только в системах с резервированием. Поскольку использование резерва увеличивает потребление ресурсов системы (процессор, оперативная память и другое), то исключение виртуальной АТС из схемы резервирования позволяет сэкономить часть ресурсов и направить сэкономленные ресурсы на обработку вызовов. В штатной работе системы это позволяет увеличить производительность в ущерб надежности. |
| callcenter\enabled | true | Доступ к контакт-центру для данной виртуальной АТС. |
| callcenter\active_agents | infinity (ограничено лицензией) | Максимальное количество подключаемых агентов Call-центра для домена. |
| callcenter\active_supervisors | infinity (ограничено лицензией) | Максимальное количество подключаемых супервизоров Call-центра для домена. |
| tc\active_conferences | infinity (ограничено лицензией) | Максимальное количество активных конференций для домена. |
| tc_count_active_channels | infinity (ограничено лицензией) | Максимальное количество подключаемых абонентов в конференцию сервиса Teleconference для домена. |
| ivr\enabled | true | Доступ к функциям IVR и dialer для данной виртуальной АТС. |
| ivr\incoming_script\enabled | true | Использовать для входящих транков в качестве контекста маршрутизации IVR-скрипт default_incoming_call. |
| teleconference\enabled | true | Доступ к сервису "Селекторная связь" для данной виртуальной АТС. |
| tsmn\concurrent_calls | 0 | Общее количество одновременно активных вызовов для системы TSMN на основном транке. |
| tsmn\concurrent_calls\redundancy | 0 | Общее количество одновременно активных вызовов для системы TSMN на резервном транке. |
| add_on_conferences_limit | infinity (ограничено лицензией) | Общее количество одновременно активных конференций для данной виртуальной АТС. |
| meet_me_limit | infinity (ограничено лицензией) | Общее количество активных пользователей "meet me" комнат для данной виртуальной АТС. |

| Название свойства | Значение по умолчанию | Описание |
|-------------------------|---------------------------------|--|
| chat_room_limit | infinity (ограничено лицензией) | Общее количество активных конференц-комнат для данной виртуальной АТС. |
| dialer\channels | 0 (ограничено лицензией) | Количество одновременных вызовов для кампаний обзвона. |
| recorder\voice\channels | 0 (ограничено лицензией) | Количество одновременных каналов записи разговоров. |
| ss_package | 0 (ограничено лицензией) | Количество лицензионных пакетов услуг. |

Аналог команды в CoSop:

`domain/<DOMAIN>/properties/restrictions/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_limits_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean domain limits.
      Usage: http://server:9999/commands/domain\_limits\_clean.
      Implementation: hc_domain_limits_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1

Запрос: http://192.168.1.21:9999/commands/domain_limits_clean

```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="domain_lim
its_clean.xsd">
  <request>
    <domain_options>
      <system>
        <option name="failover" />
      </system>
      <domain name="ct.office">
        <option name="failover" />
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="domain_lim
its_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="failover"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_domain_limits_info

Команда для просмотра ограничений, заданных на определенном домене. Список ограничений см в описании команды [Нс_domain_limits_clean](#)

Аналог команды в CoCon:

domain/<DOMAIN>/properties/restrictions/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_limits_info

Код HTTP-ответа:

- 200 – в случае успеха;

- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show domain limits.
      Usage: http://server:9999/commands/domain\_limits\_info.
      Implementation: hc_domain_limits_info.erl.
      Git: git@git.eltex.loc:8080/httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- callType -->
  <xs:simpleType name="callType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="-1"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- domainPropertiesOptionsType -->
  <xs:complexType name="domainPropertiesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="virtual_alias_limit" type="callType" use="optional"/>
        <xs:attribute name="alias_limit" type="callType" use="optional"/>
        <xs:attribute name="call_limit" type="callType" use="optional"/>
        <xs:attribute name="add_on_conferences_limit" type="callType" use="optional"/>
        <xs:attribute name="meet_me_limit" type="callType" use="optional"/>
        <xs:attribute name="chat_room_limit" type="callType" use="optional"/>
        <xs:attribute name="digitmap" type="stringType" use="optional"/>
        <xs:attribute name="callcenter_enabled" type="booleanType" use="optional"/>
        <xs:attribute name="callcenter_active_agents" type="callType" use="optional"/>
        <xs:attribute name="callcenter_active_supervisors" type="callType" use="optional"/>
      </xs:extension>
      <xs:attribute name="teleconference_enabled" type="booleanType" use="optional"/>
      <xs:attribute name="teleconference_active_channels" type="callType" use="optional"/>
      <xs:attribute name="teleconference_active_conferences" type="callType" use="optional"/>
      <xs:attribute name="tsmn_concurrent_calls" type="positiveIntegerType" use="optional"/>
      <xs:attribute name="tsmn_concurrent_calls_redundancy" type="positiveIntegerType" use="optional"/>
      <xs:attribute name="dialer_channels" type="positiveIntegerType" use="optional"/>
      <xs:attribute name="recorder_voice_channels" type="positiveIntegerType" use="optional"/>
      <xs:attribute name="ivr_enabled" type="booleanType" use="optional"/>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>

```

```

"/>
    <xs:attribute name="ivr_incoming_script_enabled" type="booleanType" use="optional
    <xs:attribute name="failover" type="booleanType" use="optional"/>
    <xs:attribute name="ss_package" type="stringType" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Count of licence from ss package in current domain.
          Example: "Expanded 1:10\nBASIC:100\nExpanded:2:10"
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoInType" minOccurs="
1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:** http://192.168.1.21:9999/commands/domain_limits_info

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_domain_limits_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="domain_limits_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="domainPropertiesOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="domainPropertiesOptionsType" virtual_alias_limit="-1"
tsmn_concurrent_calls_redundancy="0" tsmn_concurrent_calls="0" teleconference_enabled="true"
teleconference_active_conferences="-1" teleconference_active_channels="10" ss_package="-1"
recorder_voice_channels="-1" meet_me_limit="-1" ivr_incoming_script_enabled="true" ivr_enabled="true"
failover="true" digitmap="" dialer_channels="0" chat_room_limit="-1" callcenter_enabled="true"
callcenter_active_supervisors="-1" callcenter_active_agents="20" call_limit="50" alias_limit="1000"
add_on_conferences_limit="-1"/>
      </domain>
    </domain_options>
  </result>
</out>
```

[Hc_domain_limits_set](#)

Команда изменения ограничений, заданных на определенном домене (команда доступна только для администратора ECSS-10). Список ограничений см в описании команды [Hc_domain_limits_clean](#)

Аналог команды в CoCon:

domain/<DOMAIN>/properties/restrictions/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_limits_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set domain limits.
      Usage: http://server:9999/commands/domain\_limits\_set.
      Implementation: hc_domain_properties_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- callType -->
  <xs:simpleType name="callType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="-1"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- domainPropertiesOptionsType -->
  <xs:complexType name="domainPropertiesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="virtual_alias_limit" type="callType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Total virtual aliases in the domain. In case of no limits on alias
              count virtual_alias_limit = -1.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="alias_limit" type="callType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Total aliases in the domain. In case of no limits on alias count
              alias_limit = -1.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="call_limit" type="callType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Total simultaneous active calls in the domain. In case of no limits
              on simultaneous active calls call_limit = -1.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

<xs:attribute name="chat_room_limit" type="callType" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Total simultaneous chat rooms in the domain. In case of no limits on
simultaneous chat rooms chat_room_limit = -1.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="add_on_conferences_limit" type="callType" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Total simultaneous add-on conferences in the domain. In case of no
limits on simultaneous add-on conferences add_on_conferences_limit = -1.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="meet_me_limit" type="callType" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Total simultaneous meet me conferences in the domain. In case of no
limits on simultaneous meet me meet_me_limit = -1.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="digitmap" type="stringType" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Digitmap for the specified domain
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="callcenter_enabled" type="booleanType" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Domain support callcenter services.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="callcenter_active_agents" type="callType" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Total simultaneous callcenter's active agents in the domain. In case
of no limits on simultaneous active agents callcenter_active_agents = -1.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="callcenter_active_supervisors" type="callType" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">

```

```

        Total simultaneous callcenter's active supervisors in the domain. In
        case of no limits on simultaneous active supervisors callcenter_active_supervisors = -1.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="teleconference_enabled" type="booleanType" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Domain support teleconference services.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="teleconference_active_channels" type="callType" use="optional
">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Total simultaneous teleconference's active channels in the domain. In
            case of no limits on simultaneous active channels teleconference_active_channels = -1.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="teleconference_active_conferences" type="callType" use="optio
nal">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Total simultaneous teleconferences active channels in the domain. In
            case of no limits on simultaneous active teleconferences teleconference_active_conferences = -1.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="tsmn_concurrent_calls" type="positiveIntegerType" use="option
al">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Total counts of concurrent calls on master trunk for TSMN system.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="tsmn_concurrent_calls_redundancy" type="positiveIntegerType"
use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Total counts of concurrent calls on backup trunk for TSMN system.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="dialer_channels" type="positiveIntegerType" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Channels on dialer outgoing calls.

```

```

        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="recorder_voice_channels" type="positiveIntegerType" use="opti
onal">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Total count of simultaneous records voice calls.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="ivr_enabled" type="booleanType" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Domain support IVR services.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="ivr_incoming_script_enabled" type="booleanType" use="optional
">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Domain support IVR incoming services.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="failover" type="booleanType" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Domain support reincarnate active calls in case on one of the
server's is down (Work only with cluster version on ECSS-10).
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="ss_package" type="stringType" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Count of licence from ss package in current domain.
            Example: "Expanded 1:10\nBASIC:100"
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
    <xs:all>
        <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>

```

```

</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
          <xs:attribute name="force" type="xs:boolean"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/domain_limits_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="domain_limits_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="domainPropertiesOptionsType" />
      </system>
      <domain name="biysk.local">
        <options xs:type="domainPropertiesOptionsType" alias_limit="1000" call_limit="50"
callcenter_enabled="true" teleconference_enabled="true" ivr_enabled="true" failover="true"
callcenter_active_agents="20" teleconference_active_channels="10" />
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="domain_limits_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="alias_limit"/>
        <success option="call_limit"/>
        <success option="callcenter_active_agents"/>
        <success option="callcenter_enabled"/>
        <success option="failover"/>
        <success option="ivr_enabled"/>
        <success option="teleconference_active_channels"/>
        <success option="teleconference_enabled"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_domain_pbх_create - Создание PBX-домена с предопределенными настройками

Нс_domain_pbх_create

Команда позволяет создать домен с предопределенными настройками под небольшие виртуальные АТС.

Аналог команды в CoСon:

domain/pbx-declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_pbх_create

Код HTTP-ответа:

- 200 – в случае успеха;
- 206 – в случае промежуточных ответов;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- typeType -->
  <xs:simpleType name="typeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="pbx10"/>
      <xs:enumeration value="pbx100"/>
      <xs:enumeration value="pbx1000"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- cityNumberType -->
  <xs:complexType name="cityNumberType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="city_number" type="cityNumberType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="pbx_name" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The name of the creating virtual PBX.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="type" type="typeType" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The type of the created PBX.
          pbx10 - three digits numbers, no more than 10 subscribers;
          pbx100 - three digits numbers, no more than 100 subscribers;
          pbx1000 - four digits numbers, no more than 1000 subscribers.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="default_ss_profile" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          the name of the system profile name to be copied in the creating virtual PBX.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="alias_limit" type="xs:positiveInteger" use="required">
      <xs:annotation>

```



```

        <xs:documentation xml:lang="en">
            Limit to the number of aliases.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="external_call_limit" type="xs:positiveInteger" use="required">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Limit to the number of external calls (calls from/to PBX).
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="total_call_limit" type="xs:positiveInteger" use="required">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Limit to the number of calls in PBX.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="sip_ip_set" type="xs:string" use="required">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            The name of the SIP's IP set.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="parent_pbx" type="xs:string" use="required">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            The name of the PXB to make external calls.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="parent_routing_context" type="xs:string" use="required">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            The default routing context for incoming calls from creating PBX to the
parent PBX.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="parent_np" type="xs:string" use="required">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            The name of the numbering plan in the ParentPXB for created PBX.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="force" type="xs:boolean" use="optional" default="true">
    <xs:annotation>

```

```

        <xs:documentation xml:lang="en">
            Declare PBX domain without any questions, notify messages.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
</xs:complexType>

<!-- sipUserType -->
<xs:complexType name="sipUserType">
    <xs:attribute name="uri" type="xs:string" use="required">
        <xs:annotation>
            <xs:documentation xml:lang="en">
                SIP user's URI.
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="login" type="xs:string" use="required">
        <xs:annotation>
            <xs:documentation xml:lang="en">
                SIP user's login.
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="password" type="xs:string" use="required">
        <xs:annotation>
            <xs:documentation xml:lang="en">
                SIP user's password.
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>

<!-- coconUserType -->
<xs:complexType name="coconUserType">
    <xs:attribute name="login" type="xs:string" use="required">
        <xs:annotation>
            <xs:documentation xml:lang="en">
                CoCon's user login.
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="password" type="xs:string" use="required">
        <xs:annotation>
            <xs:documentation xml:lang="en">
                CoCon's user password.
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>

```

```

<!-- resultType -->
<xs:complexType name="resultType">
  <xs:sequence>
    <xs:element name="cocon_user" type="coconUserType" minOccurs="1" maxOccurs="1"/>
    <xs:element name="sip_user" type="sipUserType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!-- errorType -->
<xs:complexType name="errorType">
  <xs:attribute name="description" type="xs:string" use="optional"/>
</xs:complexType>

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request" type="requestType"/>
    </xs:all>
  </xs:complexType>
</xs:element>

<!-- out-->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/domain_pbx_create

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_domain_pbx_create.xsd">
  <request pbx_name="pbx100_new" type="pbx10" alias_limit="10" external_call_limit="5"
  total_call_limit="7" sip_ip_set="ipset2"
    parent_pbx="biysk.local" parent_routing_context="ctx_from_local" parent_np="np_240"
  force="true">
    <city_number value="240475" />
    <city_number value="240471" />
  </request>
</in>

```

Ответ:

Код: 201

Hc_domain_properties - Управление свойствами виртуальной АТС

- [Hc_domain_properties_info](#)
- [Hc_domain_properties_set](#)
- [Hc_domain_properties_clean](#)

[Hc_domain_properties_info](#)

Команда для просмотра глобальных доменных свойств и свойств, установленных на определенном домене.

Аналог команды в CoCon:

domain/<DOMAIN>/properties/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_properties_info

Код HTTP-ответа::

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show domain properties.
      Usage: http://server:9999/commands/domain_propterties_info.
      Implementation: hc_domain_properties_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- replicaTypeType -->
  <xs:simpleType name="replicaTypeType">
    <xs:restriction base="unionType">
      <xs:enumeration value="master"/>
      <xs:enumeration value="backup"/>
      <xs:enumeration value="none"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- vatsTypeType -->
  <xs:simpleType name="vatsTypeType">
    <xs:restriction base="unionType">
      <xs:enumeration value="private"/>
      <xs:enumeration value="local"/>
      <xs:enumeration value="international"/>
      <xs:enumeration value="transit"/>
      <xs:enumeration value="transit_offroad"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- defaultCallPolicyType -->
  <xs:simpleType name="defaultCallPolicyType">
    <xs:restriction base="unionType">
      <xs:enumeration value="allow"/>
      <xs:enumeration value="deny"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- redirectingNumberModeType -->
  <xs:simpleType name="redirectingNumberModeType">
    <xs:restriction base="unionType">
      <xs:enumeration value="origin_b"/>
      <xs:enumeration value="modified_b"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- domainPropertiesOptionsType -->
  <xs:complexType name="domainPropertiesOptionsType">

```

```

<xs:complexContent>
  <xs:extension base="baseOptionsType">
    <xs:attribute name="ss_codes_report_template" type="stringType" use="optional"/>
    <xs:attribute name="vats_type" type="vatsTypeType" use="optional"/>
    <xs:attribute name="trunk_control_in" type="booleanType" use="optional"/>
    <xs:attribute name="trunk_control_out" type="booleanType" use="optional"/>
    <xs:attribute name="abonent_control_out" type="booleanType" use="optional"/>
    <xs:attribute name="cw_play_voice_message" type="booleanType" use="optional"/>
    <xs:attribute name="call_recording_announcement_local" type="booleanType"
use="optional"/>
    <xs:attribute name="call_recording_announcement_external" type="booleanType"
use="optional"/>
    <xs:attribute name="default_call_policy" type="defaultCallPolicyType"
use="optional"/>
    <xs:attribute name="redirecting_number_mode" type="redirectingNumberModeType"
use="optional"/>
    <xs:attribute name="connected_number_mode" type="redirectingNumberModeType"
use="optional"/>
    <xs:attribute name="tc_display_name" type="stringType" use="optional"/>
    <xs:attribute name="site" type="stringType" use="optional"/>
    <xs:attribute name="replica_type" type="replicaTypeType" use="optional"/>
    <xs:attribute name="tc_notification_ban_timeout" type="positiveIntegerType"
use="optional"/>
    <xs:attribute name="teleconference_notification_ban_timeout"
type="positiveIntegerType" use="optional"/>
    <xs:attribute name="public_names" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The list of domain's public names splitted by comma.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="alternate_route_acp_causes" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The list of release ACP causes for try make call at alternate trunk/
direction splitted by comma.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="alternate_route_isup_causes" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The list of release ISUP causes for try make call at alternate trunk/
direction splitted by comma.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="alternate_route_sip_causes" type="xs:string" use="optional">
      <xs:annotation>

```

```

        <xs:documentation xml:lang="en">
            The list of release SIP causes for try make call at alternate trunk/
direction splitted by comma.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="dialer_repeat_acp_causes" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            The list of release ACP causes for repeat call attempt for this
causes splitted by comma.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="call_record_server_peer" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            RestFS server peer for Call Record service.
            Format: system | URI
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="call_forwarding_isup_causes" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            The list of release ISUP causes for continue original call splitted
by comma.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="voicemail_server_peer" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            RestFS server peer for Voicemail service.
            Format: system | URI
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="fax_server_peer" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            RestFS server peer for Fax receiver.
            Format: system | URI
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="sounds_server_peer" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            RestFS server peer for sounds.

```

```

        Format: system | URI
    </xs:documentation>
</xs:annotation>
</xs:attribute>
<xs:attribute name="speech_recognition_server_peer" type="xs:string"
use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            RestFS server peer for speech recognition.
            Format: system | URI
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="pictures_server_peer" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            RestFS server peer for pictures.
            Format: system | URI
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
    <xs:all>
        <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>

```



```
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример 1:

Запрос: http://192.168.1.21:9999/commands/domain_properties_info

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_domain_properties_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>
```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_properties_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="domainPropertiesOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="domainPropertiesOptionsType" voicemail_server_peer="system"
          vats_type="private" trunk_control_out="true" trunk_control_in="true"
          teleconference_notification_ban_timeout="1000" tc_notification_ban_timeout="1000"
          tc_display_name="Teleconference" ss_codes_report_template="default"
          speech_recognition_server_peer="system" sounds_server_peer="system" site="ecss10"
          replica_type="none" redirecting_number_mode="modified_b" public_names="biysk.local"
          pictures_server_peer="system" fax_server_peer="system"
          dialer_repeat_acp_causes="calledPartyRejected,noCircuitAvailable,noRequestedCircuitAvailable"
          default_call_policy="allow" cw_play_voice_message="true" connected_number_mode="origin_b"
          call_recording_announcement_local="true" call_recording_announcement_external="true"
          call_record_server_peer="system" call_forwarding_isup_causes="21,25"
          alternate_route_sip_causes="" alternate_route_isup_causes=""
          alternate_route_acp_causes=
            "routeFailure1,routeFailure2,routeSelectFailure,noCircuitAvailable,noRequestedCircuitAvailable,te
            rminationDenied,notReachable" abonent_control_out="true"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Пример 2:

Запрос: http://192.168.1.21:9999/commands/domain_properties_info

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_domain_properties_info.xsd">
  <request>
    <domain_options/>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_properties_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="domainPropertiesOptionsType" voicemail_server_peer="system"
          vats_type="private" trunk_control_out="true" trunk_control_in="true"
          teleconference_notification_ban_timeout="1000" tc_notification_ban_timeout="1000"
          tc_display_name="Teleconference" ss_codes_report_template="default"
          speech_recognition_server_peer="system" sounds_server_peer="system" site="ecss10"
          replica_type="none" redirecting_number_mode="modified_b" public_names=""
          pictures_server_peer="system" fax_server_peer="system"
          dialer_repeat_acp_causes="calledPartyRejected,noCircuitAvailable,noRequestedCircuitAvailable"
          default_call_policy="allow" cw_play_voice_message="true" connected_number_mode="origin_b"
          call_recording_announcement_local="false" call_recording_announcement_external="false"
          call_record_server_peer="system" call_forwarding_isup_causes="21,25"
          alternate_route_sip_causes="" alternate_route_isup_causes=""
          alternate_route_acp_causes=
            "routeFailure1,routeFailure2,routeSelectFailure,noCircuitAvailable,noRequestedCircuitAvailable,te
            rminationDenied,notReachable" abonent_control_out="true"/>
        </system>
        <domain name="test.domain">
          <options xs:type="domainPropertiesOptionsType" site="ecss10"/>
        </domain>
        <domain name="ct.office">
          <options xs:type="domainPropertiesOptionsType" site="ecss10"/>
        </domain>
        <domain name="biysk.local">
          <options xs:type="domainPropertiesOptionsType" site="ecss10"
            public_names="biysk.local" call_recording_announcement_local="true"
            call_recording_announcement_external="true"/>
        </domain>
        <domain name="autotest">
          <options xs:type="domainPropertiesOptionsType" site="ecss10"/>
        </domain>
      </domain_options>
    </result>
  </out>

```

hc_domain_properties_set

Команда для редактирования глобальных доменных свойств и свойств, установленных на определенном домене.

Аналог команды в CoCon:

domain/<DOMAIN>/properties/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_properties_set

Код HTTP-ответа::

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set domain properties.
      Usage: http://server:9999/commands/domain_properties_set.
      Implementation: hc_domain_properties_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- replicaTypeType -->
  <xs:simpleType name="replicaTypeType">
    <xs:restriction base="unionType">
      <xs:enumeration value="master"/>
      <xs:enumeration value="backup"/>
      <xs:enumeration value="none"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- vatsTypeType -->
  <xs:simpleType name="vatsTypeType">
    <xs:restriction base="unionType">
      <xs:enumeration value="private"/>
      <xs:enumeration value="local"/>
      <xs:enumeration value="international"/>
      <xs:enumeration value="transit"/>
      <xs:enumeration value="transit_offroad"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- defaultCallPolicyType -->
  <xs:simpleType name="defaultCallPolicyType">
    <xs:restriction base="unionType">
      <xs:enumeration value="allow"/>
      <xs:enumeration value="deny"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- redirectingNumberModeType -->
  <xs:simpleType name="redirectingNumberModeType">
    <xs:restriction base="unionType">
      <xs:enumeration value="origin_b"/>
      <xs:enumeration value="modified_b"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- domainPropertiesOptionsType -->
  <xs:complexType name="domainPropertiesOptionsType">

```

```

<xs:complexContent>
  <xs:extension base="baseOptionsType">
    <xs:attribute name="ss_codes_report_template" type="stringType" use="optional"/>
    <xs:attribute name="vats_type" type="vatsTypeType" use="optional"/>
    <xs:attribute name="trunk_control_in" type="booleanType" use="optional"/>
    <xs:attribute name="trunk_control_out" type="booleanType" use="optional"/>
    <xs:attribute name="abonent_control_out" type="booleanType" use="optional"/>
    <xs:attribute name="cw_play_voice_message" type="booleanType" use="optional"/>
    <xs:attribute name="call_recording_announcement_local" type="booleanType"
use="optional"/>
    <xs:attribute name="call_recording_announcement_external" type="booleanType"
use="optional"/>
    <xs:attribute name="default_call_policy" type="defaultCallPolicyType"
use="optional"/>
    <xs:attribute name="redirecting_number_mode" type="redirectingNumberModeType"
use="optional"/>
    <xs:attribute name="connected_number_mode" type="redirectingNumberModeType"
use="optional"/>
    <xs:attribute name="tc_display_name" type="stringType" use="optional"/>
    <xs:attribute name="site" type="stringType" use="optional"/>
    <xs:attribute name="replica_type" type="replicaTypeType" use="optional"/>
    <xs:attribute name="tc_notification_ban_timeout" type="positiveIntegerType"
use="optional"/>
    <xs:attribute name="teleconference_notification_ban_timeout"
type="positiveIntegerType" use="optional"/>
    <xs:attribute name="public_names" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The list of domain's public names splitted by comma.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="alternate_route_acp_causes" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The list of release ACP causes for try make call at alternate trunk/
direction splitted by comma.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="alternate_route_isup_causes" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The list of release ISUP causes for try make call at alternate trunk/
direction splitted by comma.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="alternate_route_sip_causes" type="xs:string" use="optional">
      <xs:annotation>

```

```

        <xs:documentation xml:lang="en">
            The list of release SIP causes for try make call at alternate trunk/
direction splitted by comma.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="dialer_repeat_acp_causes" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            The list of release ACP causes for repeat call attempt for this
causes splitted by comma.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="call_forwarding_isup_causes" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            The list of release ISUP causes for continue original call splitted
by comma.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="call_record_server_peer" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            RestFS server peer for Call Record service.
            Format: system | URI
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="voicemail_server_peer" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            RestFS server peer for Voicemail service.
            Format: system | URI
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="fax_server_peer" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            RestFS server peer for Fax receiver.
            Format: system | URI
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="sounds_server_peer" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            RestFS server peer for sounds.

```

```

        Format: system | URI
    </xs:documentation>
</xs:annotation>
</xs:attribute>
<xs:attribute name="speech_recognition_server_peer" type="xs:string"
use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            RestFS server peer for speech recognition.
            Format: system | URI
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="pictures_server_peer" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            RestFS server peer for pictures.
            Format: system | URI
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType -->
<xs:complexType name="resultType">
    <xs:all>
        <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
</xs:complexType>
<!-- in -->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainSetInType"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>

```



```

        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Запрос: http://192.168.1.21:9999/commands/domain_properties_set

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_domain_properties_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="domainPropertiesOptionsType" vats_type="private" />
      </system>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_properties_set.xsd">
  <result>
    <domain_options>
      <system>
        <success option="vats_type"/>
      </system>
    </domain_options>
  </result>
</out>

```

Пример 2:

Запрос: http://192.168.1.21:9999/commands/domain_properties_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_properties_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="domainPropertiesOptionsType" vats_type="international" />
      </system>
      <domain name="ct.office">
        <options xs:type="domainPropertiesOptionsType" vats_type="transit" />
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_properties_set.xsd">
  <result>
    <domain_options>
      <system>
        <success option="vats_type"/>
      </system>
      <domain name="ct.office">
        <success option="vats_type"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Пример 3:

Запрос: http://192.168.1.21:9999/commands/domain_properties_set

```
<?xml version='1.0' encoding='latin1'?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request>
    <domain_options>
      <system>
        <options xs:type="domainPropertiesOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options alternate_route_acp_causes="routeFailure1,routeFailure2,routeSelectFailure,
noCircuitAvailable,noRequestedCircuitAvailable,terminationDenied,notReachable/system,
conversationTimeout,ss7Failure" xs:type="domainPropertiesOptionsType"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_properties_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="alternate_route_acp_causes"/>
      </domain>
    </domain_options>
  </result>
</out>
```

[Hc_domain_properties_clean](#)

Команда для сброса в значение по умолчанию глобальных доменных свойств и свойств, установленных на определенном домене.

Аналог команды в CoCon:

domain/<DOMAIN>/properties/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_properties_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean domain properties.
      Usage: http://server:9999/commands/domain_properties_clean.
      Implementation: hc_domain_properties_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

Запрос: http://192.168.1.21:9999/commands/domain_properties_clean

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_properties_clean.xsd">
  <request>
    <domain_options>
      <system>
        <option name="vats_type"/>
      </system>
    </domain_options>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_properties_clean.xsd">
  <result>
    <domain_options>
      <system>
        <success option="vats_type"/>
      </system>
    </domain_options>
  </result>
</out>
```

Пример 2:

Запрос: http://192.168.1.21:9999/commands/domain_properties_clean

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_properties_clean.xsd">
  <request>
    <domain_options>
      <system>
        <option name="vats_type" />
      </system>
      <domain name="ct.office">
        <option name="vats_type" />
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_properties_clean.xsd">
  <result>
    <domain_options>
      <system>
        <success option="vats_type"/>
      </system>
      <domain name="ct.office">
        <success option="vats_type"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Пример 4:

Нс_domain_radius_server - Настройка RADIUS серверов на уровне домена

- [Нс_domain_radius_servers_info](#)
- [Нс_domain_radius_servers_set](#)

Нс_domain_radius_servers_info

Команда позволяет получить информацию о настройках RADIUS серверов на уровне домена.

Аналог команды в CoCon:

`/domain/<DOMAIN>/aaa/radius/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_radius_servers_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show domain RADIUS servers settings.
      Usage: http://server:9999/commands/domain\_radius\_servers\_info.
      Implementation: hc_domain_radius_servers_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="radiusOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="servers" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              The list of the RADIUS servers' settings splitted by semicolon.
              id - RADIUS server id;
              host - IP or host name of server;
              port - port number [1-49151];
              secret - RADIUS server security key;
              retry-count - number of repeats when no answer from RADIUS server (>=
0);
              retry-timeout - timeout for RADIUS server response (> 50 ms);
              idle-timeout - pause for sending requests to unaccessible RADIUS
server (>= 0 ms).

              Sample: id:master,host:127.0.0.1,port:1812,secret:test123,retry-
count:3,retry-timeout:1000,idle-timeout:60000;
              id:backup,host:127.0.0.1,port:1813,secret:test123,retry-count:
3,retry-timeout:1000,idle-timeout:60000;
              id:backup,host:::1,port:1813,secret:test123,retry-count:3,retry-
timeout:1000,idle-timeout:60000,ip-family:inet6
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>

```



```

    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainInfoInType" minOccurs="
1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/domain_radius_servers_info

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="domain_r
adius_servers_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="domain_rad
  ius_servers_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="radiusOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="radiusOptionsType" servers="id:master,host:192.168.2.12,port:
        1813,secret:eltex_secret2,retry-count:3,retry-timeout:100,idle-timeout:5000;id:backup,host:
        192.168.2.14,port:1813,secret:eltex_secret,retry-count:3,retry-timeout:1000,idle-timeout:0"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_domain_radius_servers_set

Изменение настроек RADIUS серверов на уровне домена (включая создание новых RADIUS серверов).

Аналог команды в CoCon:

`/domain/<DOMAIN>/aaa/radius/set`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_radius_servers_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set domain RADIUS servers settings.
      Usage: http://server:9999/commands/domain\_radius\_servers\_set.
      Implementation: hc_domain_radius_servers_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="radiusOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="servers" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              The list of the RADIUS servers' settings splitted by semicolon.
              id - RADIUS server id;
              host - IP or host name of server;
              port - port number [1-49151];
              secret - RADIUS server security key;
              retry-count - number of repeats when no answer from RADIUS server (>=
0);
              retry-timeout - timeout for RADIUS server response (> 50 ms);
              idle-timeout - pause for sending requests to unaccessible RADIUS
server (>= 0 ms).

              Sample: id:master,host:127.0.0.1,port:1812,secret:test123,retry-
count:3,retry-timeout:1000,idle-timeout:60000;
              id:backup,host:127.0.0.1,port:1813,secret:test123,retry-count:
3,retry-timeout:1000,idle-timeout:60000;
              id:backup,host:::1,port:1813,secret:test123,retry-count:3,retry-
timeout:1000,idle-timeout:60000,ip-family:inet6
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>

```

```

</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/domain_radius_servers_set

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="domain_
radius_servers_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="radiusOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="radiusOptionsType" servers="id:master,host:192.168.2.12,port:
1813,secret:eltex_secret2,retry-count:3,retry-timeout:100,idle-timeout:5000;id:backup,host:
192.168.2.14,port:1813,secret:eltex_secret,retry-count:3,retry-timeout:1000,idle-timeout:0"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="domain_radi
  us_servers_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="servers"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Hc_domain_regime - Управление режимами обслуживания абонентов

- [Hc_domain_import_regime](#)
- [Hc_domain_regime_info](#)
- [Hc_domain_regime_remove](#)
- [Hc_domain_regime_declare](#)

Hc_domain_import_regime

Команда позволяет изменить параметр Regime у абонентов системы при помощи списка. Команда используется для оперативного изменения режима обслуживания у абонентов.

Аналог команды в CoCon:

```
/cluster/storage/<CLUSTER>/alias/import-regime
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_import_regime

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 206 – в случае прогресса;
- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="regimeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="domainType">
    <xs:sequence>
      <xs:element name="regime" type="regimeType" maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="refType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="okType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="regime" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="errorType">
    <xs:attribute name="reason" type="xs:string" use="required"/>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="regime" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="domain" type="domainType" minOccurs="1" />
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="progress">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="ref" minOccurs="1" maxOccurs="1" type="refType" />
        <xs:element name="ok" minOccurs="0" maxOccurs="unbounded" type="okType" />
        <xs:element name="error" minOccurs="0" maxOccurs="unbounded" type="errorType" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: `http://192.168.1.21:9999/commands/domain_import_regime`

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_import_regime.xsd">
  <domain name="biysk.local">
    <regime name="accpori" address="240500" />
    <regime name="accpori" address="240501" />
    <regime name="accpori" address="240502" />
    <regime name="block_international" address="240316" />
    <regime name="accsusp" address="240699" />
  </domain>
</in>
```

Ответ: 206

Команда посылает прогрессы с состояниями. В одном не более 10 адресов.

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_import_regime.xsd">
  <ref value="md1@ecss1:1827"/>
  <ok domain="biysk.local" address="240699" regime="accsusp"/>
  <ok domain="biysk.local" address="240316" regime="block_international"/>
  <ok domain="biysk.local" address="240502" regime="accpori"/>
  <ok domain="biysk.local" address="240501" regime="accpori"/>
  <ok domain="biysk.local" address="240500" regime="accpori"/>
</out>
```

Ис_domain_regime_info

Команда позволяет просмотреть профили Regime-доступа.

Аналог команды в CoCon:

`/domain/<DOMAIN>/regime/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/domain_regime_info`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="regimeInType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="regimeType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="private_in" type="xs:boolean" use="required"/>
    <xs:attribute name="private_out" type="xs:boolean" use="required"/>
    <xs:attribute name="local_in" type="xs:boolean" use="required"/>
    <xs:attribute name="local_out" type="xs:boolean" use="required"/>
    <xs:attribute name="zone_in" type="xs:boolean" use="required"/>
    <xs:attribute name="zone_out" type="xs:boolean" use="required"/>
    <xs:attribute name="intercity_in" type="xs:boolean" use="required"/>
    <xs:attribute name="intercity_out" type="xs:boolean" use="required"/>
    <xs:attribute name="international_in" type="xs:boolean" use="required"/>
    <xs:attribute name="international_out" type="xs:boolean" use="required"/>
    <xs:attribute name="emergency_in" type="xs:boolean" use="required"/>
    <xs:attribute name="emergency_out" type="xs:boolean" use="required"/>
    <xs:attribute name="cause" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="regime" type="regimeInType" minOccurs="1" />
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="regime" type="regimeType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```


Пример:

Запрос: http://192.168.1.21:9999/commands/domain_regime_info

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_regime_info.xsd">
  <regime domain="biysk.local" />
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_regime_info.xsd">
  <regime domain="biysk.local" name="accsusp" private_in="false" private_out="false"
local_in="false" local_out="false" zone_in="false" zone_out="false" intercity_in="false"
intercity_out="false" international_in="false" international_out="false" emergency_in="true"
emergency_out="true" cause="21" description="accsusp"/>
  <regime domain="biysk.local" name="city" private_in="true" private_out="true" local_in="true"
local_out="true" zone_in="true" zone_out="false" intercity_in="true" intercity_out="false"
international_in="true" international_out="false" emergency_in="true" emergency_out="true"
cause="21" description="block 8"/>
  <regime domain="biysk.local" name="block_international" private_in="true" private_out="true"
local_in="true" local_out="true" zone_in="true" zone_out="true" intercity_in="true"
intercity_out="true" international_in="false" international_out="false" emergency_in="true"
emergency_out="true" cause="" description=""/>
</out>
```

Ис_domain_regime_remove

Команда позволяет удалить профиль Regime-доступа.

Аналог команды в CoCon:

`/domain/<DOMAIN>/regime/remove`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_regime_remove

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="regimeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="regime" type="regimeType" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_regime_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_regime_remove.xsd">
  <regime domain="biysk.local" name="city" />
</in>
```

Ответ: 204

[Hc_domain_regime_declare](#)

Команда предназначена для создания нового профиля режима доступа.

Аналог команды в CoCon:

`/domain/<DOMAIN>/regime/declare`

Метод http запроса:

POST

Шаблон http запроса:

`http://host:port/commands/domain_regime_declare`

Код http ответа:

204 – в случае успеха;
404 – в случае ошибки.

Xsd схема xml файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="regimeType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="private_in" type="xs:boolean" use="required"/>
    <xs:attribute name="private_out" type="xs:boolean" use="required"/>
    <xs:attribute name="local_in" type="xs:boolean" use="required"/>
    <xs:attribute name="local_out" type="xs:boolean" use="required"/>
    <xs:attribute name="zone_in" type="xs:boolean" use="required"/>
    <xs:attribute name="zone_out" type="xs:boolean" use="required"/>
    <xs:attribute name="intercity_in" type="xs:boolean" use="required"/>
    <xs:attribute name="intercity_out" type="xs:boolean" use="required"/>
    <xs:attribute name="international_in" type="xs:boolean" use="required"/>
    <xs:attribute name="international_out" type="xs:boolean" use="required"/>
    <xs:attribute name="emergency_in" type="xs:boolean" use="required"/>
    <xs:attribute name="emergency_out" type="xs:boolean" use="required"/>
    <xs:attribute name="cause" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="regime" type="regimeType" minOccurs="1" maxOccurs="unbounded"
/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_regime_declare

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="../priv/
xsd/domain_regime_declare.xsd">
  <regime domain="biysk.local" name="block_international" private_in="true" private_out="true"
local_in="true" local_out="true" zone_in="true" zone_out="true"
  intercity_in="true" intercity_out="true" international_in="false"
international_out="false" emergency_in="true" emergency_out="true"/>
  <regime domain="biysk.local" name="city" private_in="true" private_out="true" local_in="true"
local_out="true" zone_in="true" zone_out="false"
  intercity_in="true" intercity_out="false" international_in="true"
international_out="false" emergency_in="true" emergency_out="true" cause="21" description="block
8"/>
</in>
```

Ответ: 204

Hc_domain_statistics - Управление системой сбора статистики для виртуальной АТС

- [Hc_domain_statistics_add](#)
- [Hc_domain_statistics_delete](#)
- [Hc_domain_statistics_statmodinfo](#)
- [Hc_domain_statistics_addcolmap](#)
- [Hc_domain_statistics_delcolmap](#)
- [Hc_domain_statistics_list](#)

[Hc_domain_statistics_add](#)

Команда позволяет добавить новую задачу по сбору статистики для домена (Виртуальной АТС).

Аналог команды в CoCon:

/domain/<domain>/statistics/add

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_statistics_add

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name = "moduleType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="e502Type1"/>
      <xs:enumeration value="e502Type10"/>
      <xs:enumeration value="e502Type11"/>
      <xs:enumeration value="e502Type12"/>
      <xs:enumeration value="e502Type15"/>
      <xs:enumeration value="e502Type16"/>
      <xs:enumeration value="e502Type2"/>
      <xs:enumeration value="e502Type20"/>
      <xs:enumeration value="e502Type3"/>
      <xs:enumeration value="e502Type4"/>
      <xs:enumeration value="e502Type5"/>
      <xs:enumeration value="e502Type6"/>
      <xs:enumeration value="e502Type9"/>
      <xs:enumeration value="hostResources"/>
      <xs:enumeration value="mgTraffic"/>
      <xs:enumeration value="q752Table1"/>
      <xs:enumeration value="q752Table15"/>
      <xs:enumeration value="q752Table2"/>
      <xs:enumeration value="q752Table3"/>
      <xs:enumeration value="q752Table4"/>
      <xs:enumeration value="q752Table5"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name = "exportNameType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="txt_file_export"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="exportType">
    <xs:attribute name="name" type="xs:string" use="required" />
    <xs:attribute name="value" type="xs:string" use="required" />
  </xs:complexType>

  <xs:complexType name="filterType">
    <xs:attribute name="name" type="xs:string" use="required" />
  </xs:complexType>

  <xs:complexType name="domainType">
    <xs:sequence>
      <xs:element name="export" type="exportType" minOccurs="0" maxOccurs="unbounded" />
      <xs:element name="filter" type="filterType" minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

```

```

<xs:attribute name="module" type="moduleType" use="required"/>
<xs:attribute name="interval" type="xs:integer" use="required"/>
<xs:attribute name="export" type="exportNameType" use="required"/>
</xs:complexType>

<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="domain" type="domainType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_statistics_add

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_statistics_add.xsd">
  <domain name="biysk.local" module="e502Type9" interval="10" export="txt_file_export">
    <export name="dir" value="/var/lib/ecss/statistics" />
    <filter name="biysk.local" />
  </domain>
</in>

```

Ответ: 201

Ис_ domain_statistics_delete

Команда позволяет удалить задачи по сбору статистики в домене.

Аналог команды в CoCon:

/domain/<domain>/statistics/delete

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_statistics_delete

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="domain" type="domainType" minOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_statistics_delete

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_statistics_delete.xsd">
  <domain name="biysk.local" id="30363236-3063-3662-3365-353432316430"/>
</in>
```

Нс_domain_statistics_statmodinfo

Команда позволяет просмотреть информацию по маппированию колонок в домене.

Аналог команды в CoCon:

/domain/<domain>/statistics/statmodinfo[-m]

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_statistics_statmodinfo

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="module" type="xs:string" use="required"/>
    <xs:attribute name="flag" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="column1Type">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="xs:string" use="required"/>
    <xs:attribute name="constrait" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="columnsType">
    <xs:sequence>
      <xs:element name="column" type="column1Type" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="column2Type">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="new_name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="columns2Type">
    <xs:sequence>
      <xs:element name="column" type="column2Type" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="columns" type="columnsType" />
        <xs:element name="m_columns" type="columns2Type" />
      </xs:choice>
    </xs:complexType>
  </xs:element>

```



```
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_statistics_statmodinfo

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_statistics_statmodinfo.xsd">
  <domain name="biysk.local" module="e502Type1" flag="-m"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_statistics_statmodinfo.xsd">
  <m_columns>
    <column name="notRouted" new_name="bad_route"/>
  </m_columns>
</out>
```

Запрос: http://192.168.1.21:9999/commands/domain_statistics_statmodinfo

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_statistics_statmodinfo.xsd">
  <domain module="e502Type1" name="biysk.local" />
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="domain_statistics_statmodinfo.xsd">
  <columns description="Overall measurements on originating traffic">
    <column name="seizures" type="integer" constraint="undefined" description="Number of
originating seizures"/>
    <column name="notRouted" type="integer" constraint="undefined" description="Number of call
attempts not routed due to: no dialling, incomplete dialling, invalid address"/>
    <column name="congestion" type="integer" constraint="undefined" description="Number of
call attempts lost due to internal congestion"/>
  </columns>
</out>
```

[Hc_domain_statistics_addcolmap](#)

Команда позволяет добавить маппирование колонок в домен.

Аналог команды в CoCon:

`/domain/<domain>/statistics/addcolmap`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_statistics_addcolmap

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name = "moduleType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="e502Type1"/>
      <xs:enumeration value="e502Type10"/>
      <xs:enumeration value="e502Type11"/>
      <xs:enumeration value="e502Type12"/>
      <xs:enumeration value="e502Type15"/>
      <xs:enumeration value="e502Type16"/>
      <xs:enumeration value="e502Type2"/>
      <xs:enumeration value="e502Type20"/>
      <xs:enumeration value="e502Type3"/>
      <xs:enumeration value="e502Type4"/>
      <xs:enumeration value="e502Type5"/>
      <xs:enumeration value="e502Type6"/>
      <xs:enumeration value="e502Type9"/>
      <xs:enumeration value="hostResources"/>
      <xs:enumeration value="mgTraffic"/>
      <xs:enumeration value="q752Table1"/>
      <xs:enumeration value="q752Table15"/>
      <xs:enumeration value="q752Table2"/>
      <xs:enumeration value="q752Table3"/>
      <xs:enumeration value="q752Table4"/>
      <xs:enumeration value="q752Table5"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="module" type="moduleType" use="required"/>
    <xs:attribute name="column" type="xs:string" use="required"/>
    <xs:attribute name="new_column" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="domain" type="domainType" minOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_statistics_addcolmap

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_statistics_addcolmap.xsd">
  <domain name="biysk.local" module="e502Type1" column="notRouted" new_column="bad_route" />
</in>
```

Нс_domain_statistics_delcolmap

Команда позволяет удалить маппирование колонок в домене.

Аналог команды в CoCon:

/domain/<domain/statistics/delcolmap

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_statistics_delcolmap

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="module" type="xs:string" use="required"/>
    <xs:attribute name="column" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="domain" type="domainType" minOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_statistics_delcolmap

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_statistics_delcolmap.xsd">
  <domain name="biysk.local" module="e502Type1" column="notRouted"/>
</in>
```

Нс_domain_statistics_list

Команда позволяет получить активные задачи по сбору статистики в рамках домена.

Аналог команды в CoCon:

/domain/<domain>/statistics/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_statistics_list

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:simpleType name = "moduleType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="e502Type1"/>
      <xs:enumeration value="e502Type10"/>
      <xs:enumeration value="e502Type11"/>
      <xs:enumeration value="e502Type12"/>
      <xs:enumeration value="e502Type15"/>
      <xs:enumeration value="e502Type16"/>
      <xs:enumeration value="e502Type2"/>
      <xs:enumeration value="e502Type20"/>
      <xs:enumeration value="e502Type3"/>
      <xs:enumeration value="e502Type4"/>
      <xs:enumeration value="e502Type5"/>
      <xs:enumeration value="e502Type6"/>
      <xs:enumeration value="e502Type9"/>
      <xs:enumeration value="hostResources"/>
      <xs:enumeration value="mgTraffic"/>
      <xs:enumeration value="q752Table1"/>
      <xs:enumeration value="q752Table15"/>
      <xs:enumeration value="q752Table2"/>
      <xs:enumeration value="q752Table3"/>
      <xs:enumeration value="q752Table4"/>
      <xs:enumeration value="q752Table5"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="idType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="module" type="moduleType" use="required"/>
    <xs:attribute name="interval" type="xs:integer" use="required"/>
    <xs:attribute name="start_time" type="xs:string" use="required"/>
    <xs:attribute name="end_time" type="xs:string" use="required"/>
    <xs:attribute name="exports" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="id" type="idType" minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_statistics_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_statistics_list.xsd">
  <domain name="biysk.local" />
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="domain_statistics_list.xsd">
  <id name="30363236-3063-3461-3330-353931636432" module="e502Type1" interval="900"
start_time="2019/11/18 11:12:19" end_time="undefined" exports="txt_file_export"/>
  <id name="30363236-3063-3236-6137-356463636332" module="e502Type9" interval="600"
start_time="2019/11/18 11:02:50" end_time="undefined" exports="txt_file_export"/>
  <id name="30363236-3063-3733-6561-386564346630" module="e502Type2" interval="900"
start_time="2019/11/18 11:23:26" end_time="undefined" exports="txt_file_export"/>
  <id name="30363236-3063-3662-3365-353432316430" module="e502Type2" interval="900"
start_time="2019/11/18 11:21:07" end_time="undefined" exports="txt_file_export"/>
  <id name="30363236-3063-3631-6661-346536313831" module="e502Type6" interval="900"
start_time="2019/11/18 11:18:39" end_time="undefined" exports="rpss_db_export"/>
  <id name="30363236-3063-3661-6635-366464323230" module="e502Type9" interval="600"
start_time="2019/11/18 11:21:03" end_time="undefined" exports="txt_file_export"/>
  <id name="30363236-3063-3535-3635-613231356430" module="hostResources" interval="900"
start_time="2019/11/18 11:15:18" end_time="undefined" exports="rpss_db_export"/>
</out>

```

Нс_domain - Управление виртуальными АТС

- [Нс_domain_declare](#)
- [Нс_list_of_domains](#)
- [Нс_domain_remove](#)
- [Нс_domain_general_info](#)

- [Hc_domain_copy](#)

Hc_domain_declare

Команда позволяет создать домен (виртуальную АТС) с возможностью задания первоначальных настроек (максимальное количество абонентов, одновременных вызовов, поддержка резервирования вызовов).

Аналог команды в CoCon:

`/domain/declare`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_declare

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="common.xsd"/>
  <xs:include schemaLocation="error.xsd"/>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="domain">
          <xs:complexType>
            <xs:attribute name="name" type="xs:string" use="required">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  The name of the creating domain.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="user" type="xs:string" use="optional">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  Obsolete attribute.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="grant_user" type="xs:boolean" default="true" use="opt
ional">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  Grant ecss-DOMAIN_NAME-domain-user privileges for current
user.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="grant_admin" type="xs:boolean" default="true" use="op
tional">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  Grant ecss-DOMAIN_NAME-domain-admin privileges for current
user.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="degraded" type="xs:boolean" default="false" use="opti
onal">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  In case of degraded = false - domain support calls failover.

```

```

        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="call_limit" type="xs:nonNegativeInteger" use="optiona
1">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Total number simultaneous calls in the current domain.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="alias_limit" type="xs:nonNegativeInteger" use="option
al">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Total number aliases that can be create in the current
domain.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="ss_profile" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            The name of the system profile to be copied in the creating
domain.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="ivr_restrictions_profile" type="xs:string" use="optio
nal">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            The name of the IVR scripts' restriction profile to be
applied for the creating domain.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
</xs:complexType>
</xs:element>
</xs:all>
</xs:complexType>
</xs:element>
<!--out-->
<xs:element name = "out">
    <xs:complexType>
        <xs:choice>
            <!--Positive case-->
            <xs:element name="response">
                <xs:complexType>
                    <xs:sequence>

```

```

        <xs:element name="ok"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--Or error-->
  <xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_declare

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_domain_declare.xsd">
  <domain name="test.biysk" call_limit="5" alias_limit="30" />
</in>

```

Ответ: 201.

hc_list_of_domains

Команда позволяет вывести список доменов (виртуальных АТС), задекларированных в системе.

Аналог команды в CoCon:

/domain/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/list_of_domains

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

domains.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- domainType-->
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="optional">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  Obsolete attribute.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="domains" minOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="0" maxOccurs="
unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/list_of_domains

```
<?xml version="1.0" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="domains.xsd" >
  <request storage="ds1"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="domains.xsd">
  <domains>
    <domain name="biysk.local"/>
    <domain name="ct.office"/>
    <domain name="d.remote"/>
  </domains>
</out>
```

Нс_domain_remove

Команда позволяет удалить домен (виртуальную АТС) из системы.

Аналог команды в CoСon:

/domain/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_remove

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="domain">
          <xs:complexType>
            <xs:attribute name="name" type="xs:string" use="required"/>
            <xs:attribute name="force" type="xs:boolean" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/domain_remove

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_domain_remove.xsd">
  <domain force="true" name="d.remote" />
</in>

```

Ответ: пустой XML.

hc_domain_general_info

Команда просмотра значений общих свойств виртуальной АТС.

Аналоги команды в CoCon (сборка):

```

/domain/<DOMAIN>/alias/subscribers-limit
/domain/<DOMAIN>/sip/user/registered
/domain/<DOMAIN>/megaco/gateways/list

```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_general_info

Код HTTP-ответа:

- 200 – в случае успеха;

- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="SubscribersInfo">
    <xs:attribute name="count" type="xs:nonNegativeInteger" use="required"/>
    <xs:attribute name="limit" type="xs:nonNegativeInteger" use="required"/>
    <xs:attribute name="general_aliases_count" type="xs:nonNegativeInteger" use="required"/>
    <xs:attribute name="virtual_aliases_count" type="xs:nonNegativeInteger" use="required"/>
    <xs:attribute name="licence_general_aliases_count" type="xs:nonNegativeInteger" use="required"/>
    <xs:attribute name="licence_virtual_aliases_count" type="xs:string" use="required"/>
    <xs:attribute name="free_general_aliases_count" type="xs:nonNegativeInteger" use="required"/>
    <xs:attribute name="free_virtual_aliases_count" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="RegisteredUsersInfo">
    <xs:attribute name="count" type="xs:nonNegativeInteger" use="required"/>
  </xs:complexType>
  <xs:complexType name="GatewayInfo">
    <xs:attribute name="count" type="xs:nonNegativeInteger" use="required"/>
  </xs:complexType>
  <xs:complexType name="DomainGeneralInfo">
    <xs:sequence>
      <xs:element name="subscribersResult">
        <xs:complexType>
          <xs:choice>
            <xs:element name="subscribersInfo" type="SubscribersInfo"/>
            <xs:element name="error" type="errorType"/>
          </xs:choice>
        </xs:complexType>
      </xs:element>
      <xs:element name="registeredUsersResult">
        <xs:complexType>
          <xs:choice>
            <xs:element name="registeredUsersInfo" type="RegisteredUsersInfo"/>
            <xs:element name="error" type="errorType"/>
          </xs:choice>
        </xs:complexType>
      </xs:element>
      <xs:element name="gatewayResult">
        <xs:complexType>
          <xs:choice>
            <xs:element name="gatewayInfo" type="GatewayInfo"/>
            <xs:element name="error" type="errorType"/>
          </xs:choice>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

```



```

</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!--Positive case-->
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domainGeneralInfo" type="DomainGeneralInfo"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <!--Or error-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/domain_general_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request domain="biysk.local"/>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_domain_
  general_info.xsd">
  <response>
    <domainGeneralInfo>
      <subscribersResult>
        <subscribersInfo count="313" limit="2000" general_aliases_count="313"
  virtual_aliases_count="0" licence_general_aliases_count="2000" licence_virtual_aliases_count="inf
  inity" free_general_aliases_count="1687" free_virtual_aliases_count="infinity"/>
      </subscribersResult>
      <registeredUsersResult>
        <registeredUsersInfo count="210"/>
      </registeredUsersResult>
      <gatewayResult>
        <gatewayInfo count="0"/>
      </gatewayResult>
    </domainGeneralInfo>
  </response>
</out>

```

hc_domain_copy

Команда предназначена для создания копии домена на основе существующего, с возможностью задать digitmap, а так же определить bridge в основной домен.

Аналог команды в CoSop:

`/domain/copy`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/domain_copy

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="common.xsd"/>
  <xs:include schemaLocation="error.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="domain">
          <xs:complexType>
            <xs:attribute name="src_name" type="xs:string" use="required">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  Source domain name.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="dst_name" type="xs:string" use="required">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  New domain name.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="digitmap" type="xs:string" use="optional">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  The digitmap for the new domain.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="src_bridge_name" type="xs:string" use="optional">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  Source bridge name (used only with dst_bridge_name).
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="dst_bridge_name" type="xs:string" use="optional">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  Destination bridge name (used only with src_bridge_name).
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

        </xs:all>
    </xs:complexType>
</xs:element>
<!--out-->
<xs:element name = "out">
    <xs:complexType>
        <xs:choice>
            <!--Positive case-->
            <xs:element name="response">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="ok"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <!--Or error-->
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/domain_copy

```

<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_domain_copy.xsd">
  <domain src_name="ct.office" dst_name="ssw.biysk2" digitmap="xxx+" src_bridge_name="br_office"
  " dst_bridge_name="bridge:bsk"/>
</in>

```

Ответ: 201

Нс_ds_cmds_list - Команды управления списками

- [Нс_ds_cmds_lists_declare](#)
- [Нс_ds_cmds_lists_info](#)
- [Нс_ds_cmds_lists_list](#)
- [Нс_ds_cmds_lists_number_add](#)
- [Нс_ds_cmds_lists_number_remove](#)
- [Нс_ds_cmds_lists_remove](#)
- [Нс_ds_cmds_lists_set](#)

Нс_ds_cmds_lists_declare

Команда создает новый список. Параметры *description* и *number* можно не задавать.

Аналог команды в CoCon:

`domain/<DOMAIN>/lists/declare`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ds_cmds_lists_declare

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Declare list of number.
      Usage: http://server:9999/commands/ds\_cmds\_lists\_declare.
      Implementation: hc_ds_cmds_lists_declare.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- numberType -->
  <xs:complexType name="numberType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- listType -->
  <xs:complexType name="listType">
    <xs:sequence>
      <xs:element name="number" type="numberType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="type" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- request -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="list" type="listType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->

```

```

<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/ds_cmds_lists_declare

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ds_cmds
_lists_declare.xsd">
  <request domain="biysk.local">
    <list name="ListName" description="NewList" type="blf" >
      <number value="240545"/>
      <number value="416977"/>
      <number value="83852352804"/>
    </list>
  </request>
</in>

```

Ответ: 201

hc_ds_cmds_lists_info

Команда предназначена для вывода указанного списка.

Аналог команды в CoCon:

`domain/<DOMAIN>/lists/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ds_cmds_lists_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show list number with description.
      Usage: http://server:9999/commands/ds\_cmds\_lists\_info.
      Implementation: hc_ds_cmds_lists_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- listTypeIn -->
  <xs:complexType name="listTypeIn">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- request -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="list" type="listTypeIn" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- numberType -->
  <xs:complexType name="numberType">
    <xs:attribute name="value" type="xs:string" use="required"/>
    <xs:attribute name="priority" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- listTypeOut -->
  <xs:complexType name="listTypeOut">
    <xs:sequence>
      <xs:element name="number" type="numberType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

```



```

    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
    <xs:attribute name="type" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:element name="list" type="listTypeOut" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="result" type="resultType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/ds_cmds_lists_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ds_cmds
_lists_info.xsd">
  <request domain="biysk.local">
    <list name="cc_list"/>
  </request>
</in>

```

Ответ

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ds_cmds
_lists_info.xsd">
  <result>
    <list name="debt_rate2" description="debt_rate2" type="default">
      <number value="416977" priority="normal"/>
      <number value="240465" priority="normal"/>
      <number value="240464" priority="normal"/>
      <number value="240101" priority="normal"/>
    </list>
  </result>
</out>
```

hc_ds_cmds_lists_list

Команда предназначена для вывода существующих списков и их описания.

Аналог команды в CoCon:

domain/<DOMAIN>/lists/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ds_cmds_lists_list

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show list of number lists.
      Usage: http://server:9999/commands/ds\_cmds\_lists\_list.
      Implementation: hc_ds_cmds_lists_list.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- request -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="type" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- listType -->
  <xs:complexType name="listType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
    <xs:attribute name="type" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:element name="list" type="listType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>

```

```

    <xs:sequence>
      <xs:element name="result" type="resultType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/ds_cmds_lists_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ds_cmds_lists_list.xsd">
  <request domain="biysk.local"/>
</in>

```

Ответ

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ds_cmds_lists_list.xsd">
  <result>
    <list name="debt_rate2" description="debt_rate2" type="default"/>
    <list name="ListName" description="NewList" type="blf"/>
  </result>
</out>

```

hc_ds_cmds_lists_number_add

Команда добавляет новые номера в существующий список.

Аналог команды в CoCon:

domain/<DOMAIN>/lists/numbers-add

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ds_cmds_lists_number_add

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Add specified numbers to exist list.
      Usage: http://server:9999/commands/ds\_cmds\_lists\_number\_add.
      Implementation: hc_ds_cmds_lists_number_add.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- numberType -->
  <xs:complexType name="numberType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- listType -->
  <xs:complexType name="listType">
    <xs:sequence>
      <xs:element name="number" type="numberType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- request -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="list" type="listType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>

```

```

    <xs:choice>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/ds_cmds_lists_number_add

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_sd_cmds_lists_number_add.xsd">
  <request domain="biysk.local">
    <list name="spam_list">
      <number value="240501"/>
      <number value="240502"/>
      <number value="240503"/>
      <number value="83852352804"/>
      <number value="83832741001"/>
      <number value="89520001615"/>
    </list>
  </request>
</in>

```

Ответ: 201

hc_ds_cmds_lists_number_remove

Команда удаляет номера из существующего списка.

Аналог команды в CoCon:

domain/<DOMAIN>/lists/numbers-remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ds_cmds_lists_number_remove

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Remove specified numbers to exist list.
      Usage: http://server:9999/commands/ds\_cmds\_lists\_number\_remove.
      Implementation: hc_ds_cmds_lists_number_remove.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- numberType -->
  <xs:complexType name="numberType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- listType -->
  <xs:complexType name="listType">
    <xs:sequence>
      <xs:element name="number" type="numberType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- request -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="list" type="listType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>

```



```

    <xs:choice>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/ds_cmds_lists_number_remove

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_sd_cmds
_lists_number_remove.xsd">
  <request domain="biysk.local">
    <list name="spam_list">
      <number value="240318"/>
      <number value="83832741001"/>
    </list>
  </request>
</in>

```

Ответ: 201

hc_ds_cmds_lists_remove

Команда удаляет существующий список.

Аналог команды в CoCon:

domain/<DOMAIN>/lists/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ds_cmds_lists_remove

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Remove specified exist list.
      Usage: http://server:9999/commands/ds\_cmds\_lists\_remove.
      Implementation: hc_ds_cmds_lists_remove.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- listType -->
  <xs:complexType name="listType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- request -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="list" type="listType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/ds_cmds_lists_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ds_cmds
_lists_remove.xsd">
  <request domain="biysk.local">
    <list name="spam_list"/>
  </request>
</in>
```

Ответ: 201

hc_ds_cmds_lists_set

Команда устанавливает новые значения для *description* и/или *numbers* в указанном списке.

Аналог команды в CoCon:

domain/<DOMAIN>/lists/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ds_cmds_lists_set

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set lists description and numbers list.
      Usage: http://server:9999/commands/ds\_cmds\_lists\_set.
      Implementation: hc_ds_cmds_lists_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- numberType -->
  <xs:complexType name="numberType">
    <xs:attribute name="value" type="xs:string" use="required"/>
    <xs:attribute name="priority" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- listType -->
  <xs:complexType name="listType">
    <xs:sequence>
      <xs:element name="number" type="numberType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="type" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- request -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="list" type="listType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/ds_cmds_lists_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ds_cmds_lists_set.xsd">
  <request domain="biysk.local">
    <list name="ListName" description="my cool spamlist" type="default">
    </list>
  </request>
</in>

```

Ответ: 201

hc_ds_domain_check_addr_to_digitmap_cmd - Проверка адреса по DigitMap'y

Команда проверяет адреса переданные в формате AddressRange на валидность по маске DigitMap определенной на домене.

Аналог команды в CoCon:

domain/<DOMAIN>/alias/.check-by-digitmap

Метод http запроса

POST

Шаблон http запроса

http://host:port/commands/ds_domain_check_addr_to_digitmap_cmd

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема xml файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="validate" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="domain" type="xs:string" use="required"/>
                  <xs:attribute name="addressrange" type="xs:string"
use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="response" minOccurs="0" maxOccurs="unbounded">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="addressrange" type="xs:string" use="required"/>
            <xs:attribute name="validate" type="xs:boolean" use="required"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

http://192.168.1.21:9999/commands/ds_domain_check_addr_to_digitmap_cmd

```
<in>
  <request>
    <validate domain="biysk.local" addressrange="240101"/>
    <validate domain="biysk.local" addressrange="2405{01-99}"/>
    <validate domain="biysk.local" addressrange="10{01-20}"/>
    <validate domain="biysk.local" addressrange="100b"/>
    <validate domain="biysk.local" addressrange="100B"/>

  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ds_domain_check_addr_to_digitmap_cmd.xsd">
  <response domain="biysk.local" addressrange="240101" validate="true"/>
  <response domain="biysk.local" addressrange="2405{01-99}" validate="true"/>
  <response domain="biysk.local" addressrange="10{01-20}" validate="true"/>
  <response domain="biysk.local" addressrange="100b" validate="false"/>
  <response domain="biysk.local" addressrange="100B" validate="true"/>
</out>
```

hc_ds_general_options - Управление свойствами DS (системные свойства)

- [Общая схема](#)
- [hc_ds_general_options_info](#)
- [hc_ds_general_options_set](#)
- [hc_ds_general_options_clean](#)

Общая схема

XSD-схема hc_ds_common.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- systemDSOptionsType -->
  <xs:complexType name="DSOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="call_limit" type="stringType" use="optional" />
        <xs:attribute name="system_restfs_peer" type="stringType" use="optional" />
        <xs:attribute name="callcenter_active_agents" type="stringType" use="optional" />
        <xs:attribute name="callcenter_active_supervisors" type="stringType" use="optional" />
        <xs:attribute name="time_difference_threshold" type="positiveIntegerType" use="optional"
      />
        <xs:attribute name="heartbeat_send_timeout" type="positiveIntegerType" use="optional" />
        <xs:attribute name="reroute_by_cause_max_hops" type="positiveIntegerType" use="optional"
      />
        <xs:attribute name="heartbeat_last_chance_timeout" type="positiveIntegerType"
      use="optional" />
        <xs:attribute name="mm_wait_ccid" type="positiveIntegerType" use="optional" />
        <xs:attribute name="mm_rebuild_topology_timeout" type="positiveIntegerType"
      use="optional" />
        <xs:attribute name="notification_services" type="positiveIntegerType" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>

```

Нс_ds_general_options_info

Команда просмотра свойств DS.

Аналог команды в CoCon:

system/info

Шаблон HTTP-запроса:

http://host:port/commands/ds_general_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="ds_general_option_info">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get options of TTS node.
      Usage: http://server:9999/commands/ds\_general\_options\_info
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <xs:include schemaLocation="hc_ds_common.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="system_options" type="systemInfoInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="system_options" type="systemInfoOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/ds_general_options_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request>
    <system_options/>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ds_gene
ral_options_info.xsd">
  <result>
    <system_options>
      <system>
        <options xs:type="DSOptionsType" call_limit="infinity" callcenter_active_agents="
infinity" callcenter_active_supervisors="infinity" system_restfs_peer="default"
time_difference_threshold="500"/>
      </system>
    </system_options>
  </result>
</out>
```

Нс_ds_general_options_set

Команда для установки свойств DS.

Аналог команды в CoCon:

system/set

Шаблон HTTP-запроса:

http://host:port/commands/ds_general_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="ds_general_options_set">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get options of TTS node.
      Usage: http://server:9999/commands/ds\_general\_options\_set
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <xs:include schemaLocation="hc_ds_common.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="system_options" type="systemSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="system_options" type="systemSetOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/ds_general_options_set

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ds_gene
ral_options_set.xsd">
  <request>
    <system_options>
      <system>
        <options xs:type="DSOptionsType" call_limit="50"/>
      </system>
    </system_options>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ds_gene
ral_options_set.xsd">
  <result>
    <system_options>
      <system>
        <success option="call_limit"/>
      </system>
    </system_options>
  </result>
</out>
```

[Hc_ds_general_options_clean](#)

Команда очистки свойств DS.

Аналог команды в CoCon:

system/clean

Шаблон HTTP-запроса:

http://host:port/commands/ds_general_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="ds_general_options_clean">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get options of TTS node.
      Usage: http://server:9999/commands/ds\_general\_options\_clean
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <xs:include schemaLocation="hc_ds_common.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="system_options" type="systemCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="system_options" type="systemCleanOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/ds_general_options_clean

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_tts_general_options_clean.xsd">
  <request>
    <system_options>
      <system>
        <option name="time_difference_threshold"/>
      </system>
    </system_options>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ds_general_options_clean.xsd">
  <result>
    <system_options>
      <system>
        <success option="time_difference_threshold"/>
      </system>
    </system_options>
  </result>
</out>
```

Hc_ecss_control_channel_options - Управление подсистемой RADIUS авторизации

- [Hc_ecss_control_channel_options_clean](#)
- [Hc_ecss_control_channel_options_info](#)
- [Hc_ecss_control_channel_options_set](#)

[Hc_ecss_control_channel_options_clean](#)

Команда сброса настроек контрольной коннекции.

Аналог команды в CoCon:

system/mediactrl/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ecss_control_channel_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean settings of control channel to MSR.
      Usage: http://server:9999/commands/ecss_control_channel_options_clean.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemCleanOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/ecss_control_channel_options_clean


```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ecss_control_channel_options_clean.xsd">
  <request>
    <system_options>
      <system>
        <option name="mediactrl_keepalive_timeout"/>
        <option name="mediactrl_transaction_timeout"/>
        <option name="msr_ban_end_timeout"/>
      </system>
    </system_options>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ecss_control_channel_options_clean.xsd">
  <result>
    <system_options>
      <system>
        <success option="msr_ban_end_timeout"/>
        <success option="mediactrl_transaction_timeout"/>
        <success option="mediactrl_keepalive_timeout"/>
      </system>
    </system_options>
  </result>
</out>
```

[Hc_ecss_control_channel_options_info](#)

Команда просмотра настроек контрольной коннекции.

Аналог команды в CoCon:

system/mediactrl/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ecss_control_channel_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show settings of control channel to MSR.
      Usage: http://server:9999/commands/ecss\_control\_channel\_options\_info.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- ControlChannelOptionsType -->
  <xs:complexType name="ControlChannelOptionsType">
    <xs:annotation>
      <xs:documentation xml:lang="en">
seconds
        mediactrl_keepalive_timeout and mediactrl_transaction_timeout - Timeout in
seconds
        msr_ban_end_timeout - Timeout in milliseconds
      </xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="mediactrl_keepalive_timeout" type="integerType"/>
        <xs:attribute name="mediactrl_transaction_timeout" type="integerType"/>
        <xs:attribute name="msr_ban_end_timeout" type="integerType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemInfoInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">

```

```

        <xs:complexType>
            <xs:all>
                <xs:element name="system_options" type="systemInfoOutType"/>
            </xs:all>
        </xs:complexType>
    </xs:element>
    <xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/ecss_control_channel_options_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ecss_control_channel_options_info.xsd">
    <request>
        <system_options/>
    </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ecss_control_channel_options_info.xsd">
    <result>
        <system_options>
            <system>
                <options xs:type="ControlChannelOptionsType" mediactrl_keepalive_timeout="10"
mediactrl_transaction_timeout="20" msr_ban_end_timeout="1000"/>
            </system>
        </system_options>
    </result>
</out>

```

[Hc_ecss_control_channel_options_set](#)

Команда изменения настроек контрольной коннекции.

Аналог команды в CoCon:

`system/mediactrl/set`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ecss_control_channel_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show settings of control channel to MSR.
      Usage: http://server:9999/commands/ecss\_control\_channel\_options\_set.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- ControlChannelOptionsType -->
  <xs:complexType name="ControlChannelOptionsType">
    <xs:annotation>
      <xs:documentation xml:lang="en">
seconds
        mediactrl_keepalive_timeout and mediactrl_transaction_timeout - Timeout in
seconds
        msr_ban_end_timeout - Timeout in milliseconds
      </xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="mediactrl_keepalive_timeout" type="integerType"/>
        <xs:attribute name="mediactrl_transaction_timeout" type="integerType"/>
        <xs:attribute name="msr_ban_end_timeout" type="integerType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">

```

```

        <xs:complexType>
            <xs:all>
                <xs:element name="system_options" type="systemSetOutType"/>
            </xs:all>
        </xs:complexType>
    </xs:element>
    <xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/ecss_control_channel_options_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ecss_control_channel_options_set.xsd">
  <request>
    <system_options>
      <system>
        <options mediactrl_keepalive_timeout="10" mediactrl_transaction_timeout="20"
msr_ban_end_timeout="1000" xs:type="ControlChannelOptionsType"/>
      </system>
    </system_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ecss_control_channel_options_set.xsd">
  <result>
    <system_options>
      <system>
        <success option="msr_ban_end_timeout"/>
        <success option="mediactrl_transaction_timeout"/>
        <success option="mediactrl_keepalive_timeout"/>
      </system>
    </system_options>
  </result>
</out>

```

Нс_ecss_mlpp_options - Управление многоадресной приоритизацией ВЫЗОВОВ

- [Нс_ecss_mlpp_options_clean](#)
- [Нс_ecss_mlpp_options_info](#)
- [Нс_ecss_mlpp_options_set](#)

[Нс_ecss_mlpp_options_clean](#)

Команда изменяет настройки на уровне виртуальной АТС или возвращает их значения по умолчанию для кластерных настроек.

Аналог команды в CoCon:

`mlpp/clean`

`domain/<DOMAIN>/mlpp/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ecss_mlpp_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean properties of MLPP service.
      Usage: http://server:9999/commands/ecss\_mlpp\_options\_clean.
      Implementation: hc_ecss_mlpp_options_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

Запрос: http://192.168.1.21:9999/commands/ecss_mlpp_options_clean

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ecss_ml
pp_options_clean.xsd">
  <request>
    <domain_options>
      <system>
        <option name="queue_time"/>
        <option name="queue_length"/>
        <option name="mode"/>
        <option name="enable"/>
      </system>
      <domain name="biysk.local">
        <option name="queue_time"/>
        <option name="mode"/>
        <option name="enable"/>
      </domain>
    </domain_options>
  </request>
</in>

```

ОТВЕТ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ecss_ml
pp_options_clean.xsd">
  <result>
    <domain_options>
      <system>
        <success option="enable"/>
        <success option="mode"/>
        <success option="queue_length"/>
        <success option="queue_time"/>
      </system>
      <domain name="biysk.local">
        <success option="enable"/>
        <success option="mode"/>
        <success option="queue_time"/>
      </domain>
    </domain_options>
  </result>
</out>

```

hc_ecss_mlpp_options_info

Команда просмотра настроек подсистемы MLPP всей системы или виртуальной АТС.

Аналог команды в CoCon:

`mlpp/info`

`domain/<DOMAIN>/mlpp/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ecss_mlpp_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of MLPP service.
      Usage: http://server:9999/commands/ecss\_mlpp\_options\_info.
      Implementation: hc_ecss_mlpp_options_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- modeType -->
  <xs:simpleType name="modeType">
    <xs:restriction base="unionType">
      <xs:enumeration value="wait"/>
      <xs:enumeration value="force"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- defaultPriorityType -->
  <xs:simpleType name="defaultPriorityType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="15"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- queueLengthType -->
  <xs:simpleType name="queueLengthType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="2"/>
      <xs:maxInclusive value="3"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- queueTimeType -->
  <xs:simpleType name="queueTimeType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="40"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- ecssMLPPOptionsType -->
  <xs:complexType name="ecssMLPPOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="default_priority" type="defaultPriorityType" use="optional"/>
        <xs:attribute name="priority" type="xs:string" use="optional"/>
        <xs:attribute name="enable" type="booleanType" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

        <xs:attribute name="mode" type="modeType" use="optional"/>
        <xs:attribute name="queue_length" type="queueLengthType" use="optional"/>
        <xs:attribute name="queue_time" type="queueTimeType" use="optional"/>
    </xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
    <xs:all>
        <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainInfoInType" minOccurs="
1"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema

```

Параметр "domain" является необязательным, если параметр не задан - возвращаются все настройки подсистемы MLPP в рамках системы.

Пример 1:

Запрос: http://192.168.1.21:9999/commands/ecss_mlpp_options_info

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_ecss_
mlpp_options_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ecss_ml
pp_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="ecssMLPPOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="ecssMLPPOptionsType" queue_time="40" queue_length="2" priority="
[]" mode="wait" enable="true" default_priority="4"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Пример 2:

Запрос: http://192.168.1.21:9999/commands/ecss_mlpp_options_info

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_ecss_
mlpp_options_info.xsd">
  <request>
    <domain_options/>
  </request>
</in>
```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ecss_ml
pp_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="ecssMLPPOptionsType" queue_time="40" queue_length="2" priority="
[]" mode="wait" enable="false" default_priority="4"/>
      </system>
      <domain name="ct.office">
        <options xs:type="ecssMLPPOptionsType" queue_length="3"/>
      </domain>
      <domain name="biysk.local">
        <options xs:type="ecssMLPPOptionsType" queue_time="25" queue_length="3" mode="for
ce" enable="false"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_ecss_mlpp_options_set

Команда настройки подсистемы MLPP для всей системы или виртуальной АТС.

Аналог команды в CoCon:

mlpp/set

domain/<DOMAIN>/mlpp/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ecss_mlpp_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set properties of MLPP service.
      Usage: http://server:9999/commands/ecss\_mlpp\_options\_set.
      Implementation: hc_ecss_mlpp_options_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- modeType -->
  <xs:simpleType name="modeType">
    <xs:restriction base="unionType">
      <xs:enumeration value="wait"/>
      <xs:enumeration value="force"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- defaultPriorityType -->
  <xs:simpleType name="defaultPriorityType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="15"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- queueLengthType -->
  <xs:simpleType name="queueLengthType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="2"/>
      <xs:maxInclusive value="3"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- queueTimeType -->
  <xs:simpleType name="queueTimeType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="40"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- ecssMLPPOptionsType-->
  <xs:complexType name="ecssMLPPOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="default_priority" type="defaultPriorityType" use="optional"/>
        <xs:attribute name="priority" type="xs:string" use="optional"/>
        <xs:attribute name="enable" type="booleanType" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>


```



```

        <xs:attribute name="mode" type="modeType" use="optional"/>
        <xs:attribute name="queue_length" type="queueLengthType" use="optional"/>
        <xs:attribute name="queue_time" type="queueTimeType" use="optional"/>
    </xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
    <xs:all>
        <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainSetInType"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

 Namespace для поля "type" в теге "options" обязательно должен быть **xs**.

Пример 1:

Запрос: http://192.168.1.21:9999/commands/ecss_mlpp_options_set

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ecss_ml
pp_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="ecssMLPPOptionsType" queue_time="40" queue_length="2" mode="wai
t" enable="false"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="ecssMLPPOptionsType" queue_time="25" mode="force" enable="false
"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ecss_ml
pp_options_set.xsd">
  <result>
    <domain_options>
      <system>
        <success option="enable"/>
        <success option="mode"/>
        <success option="queue_length"/>
        <success option="queue_time"/>
      </system>
      <domain name="biysk.local">
        <success option="enable"/>
        <success option="mode"/>
        <success option="queue_time"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Пример 2:

Запрос: http://192.168.23.34:9999/commands/ecss_mlpp_options_set

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ecss_ml
pp_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="ecssMLPPOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="ecssMLPPOptionsType" queue_length="3"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ecss_ml
pp_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="queue_length"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Нс_ecss_statistics - Получение статистики по домену

- [Нс_ecss_statistics_show](#)
- [Нс_ecss_statistics_get_metric](#)

Нс_ecss_statistics_show

Команда предназначена для получения статистики ECSS-10 за определенный интервал времени, с использованием определенных фильтров.

Аналог команды в CoCon:

/domain/<DOMAIN>/core/statistics/show

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ecss_statistics_show

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- systemType -->
  <xs:simpleType name="systemType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="ecss10.common"/>
      <xs:enumeration value="ecss10.ssw"/>
      <xs:enumeration value="ecss10.cc"/>
      <xs:enumeration value="ecss10.tc"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- statisticTypeType -->
  <xs:simpleType name="statisticTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="counter"/>
      <xs:enumeration value="gauge"/>
      <xs:enumeration value="meter"/>
      <xs:enumeration value="histogram"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- statisticsValueType -->
  <xs:complexType name="statisticsValueType">
    <xs:attribute name="time" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="value" type="xs:int"/>
    <xs:attribute name="n" type="xs:int"/>
    <xs:attribute name="min" type="xs:int" use="optional"/>
    <xs:attribute name="max" type="xs:int" use="optional"/>
    <xs:attribute name="avg" type="xs:double" use="optional"/>
  </xs:complexType>
  <!-- statisticsInType -->
  <xs:complexType name="statisticsInType">
    <xs:attribute name="from" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="to" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="system" type="systemType"/>
    <xs:attribute name="subsystem" type="xs:string" use="optional"/>
    <xs:attribute name="statistic" type="xs:string" use="optional"/>
    <xs:attribute name="type" type="statisticTypeType" use="optional"/>
    <xs:attribute name="variable_part_1" type="xs:string" use="optional"/>
    <xs:attribute name="variable_part_2" type="xs:string" use="optional"/>
    <xs:attribute name="measure_type" type="statisticIntervalType" use="optional"/>
  </xs:complexType>
  <!-- statisticsOutType -->
  <xs:complexType name="statisticsOutType">
    <xs:sequence>
      <xs:element name="value" minOccurs="0" maxOccurs="unbounded" type="statisticsValueTyp
e"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string"/>

```

```

<xs:attribute name="from" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
<xs:attribute name="to" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
<xs:attribute name="system" type="systemType"/>
<xs:attribute name="subsystem" type="xs:string"/>
<xs:attribute name="statistic" type="xs:string"/>
<xs:attribute name="type" type="statisticTypeType"/>
<xs:attribute name="measure_type" type="xs:string" use="optional"/>
<xs:attribute name="sum" type="xs:float" use="optional"/>
<xs:attribute name="variable_part_1" type="xs:string" use="optional"/>
<xs:attribute name="variable_part_2" type="xs:string" use="optional"/>
</xs:complexType>
<!-- statisticIntervalType -->
<xs:simpleType name="statisticIntervalType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="realtime"/>
    <xs:enumeration value="intrahour"/>
    <xs:enumeration value="hour"/>
    <xs:enumeration value="day"/>
    <xs:enumeration value="week"/>
    <xs:enumeration value="month"/>
    <xs:enumeration value="auto"/>
  </xs:restriction>
</xs:simpleType>
<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="statistics" maxOccurs="unbounded" type="statisticsI
nType"/>
          </xs:sequence>
          <xs:attribute name="domain" type="xs:string"/>
          <xs:attribute name="measure_type" type="statisticIntervalType" use="optio
nal"/>
          <xs:attribute name="request_time_utc" type="xs:boolean" use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>

```

```
        <xs:element name="statistics" minOccurs="0" maxOccurs="unbounded"
type="statisticsOutType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
```

Пример

Запрос: http://192.168.1.21:9999/commands/ecss_statistics_show

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ecss_statistics_show.xsd">
  <request domain="biysk.local">
    <statistics system="ecss10.ssw" subsystem="trunk_manager" from="2019/11/05 00:00:00"
to="2019/11/17 00:00:00" statistic="incoming_seizures.counter" variable_part_1="smg-4" />
  </request>
</in>
```

Ответ:

Код: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ecss_statistics_show.xsd">
  <response>
    <statistics domain="biysk.local" from="2019/11/05 00:00:00" to="2019/11/17 00:00:00"
      system="ecss10.ssw" subsystem="trunk_manager" statistic="incoming_seizures.counter"
      type="counter" measure_type="day" sum="1533.000000" variable_part_1="smg-4">
      <value time="2019/11/05 16:59:59" value="79" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/06 16:59:59" value="134" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/07 16:59:59" value="97" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/08 16:59:59" value="163" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/09 16:59:59" value="82" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/10 16:59:59" value="57" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/11 16:59:59" value="140" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/12 16:59:59" value="148" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/13 16:59:59" value="219" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/14 16:59:59" value="155" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/15 16:59:59" value="181" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/16 16:59:59" value="78" n="-" min="-" max="-" avg="-"/>
    </statistics>
  </response>
</out>

```

Пример 2

Запрос: http://192.168.23.34:9999/commands/ecss_statistics_show

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ecss_statistics_show.xsd">
  <request domain="biysk.local">
    <statistics system="ecss10.ssw" subsystem="trunk_manager" from="2019/11/05 00:00:00"
      to="2019/11/18 00:00:00" statistic="incoming_seizures.counter" variable_part_1="smg-4"/>
    <statistics system="ecss10.ssw" subsystem="trunk_manager" from="2019/11/05 00:00:00"
      to="2019/11/18 00:00:00" statistic="incoming_accepted_seizures.counter" variable_part_1="smg-4"
    />
  </request>
</in>

```

Ответ:

Код: 200


```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ecss_statistics_show.xsd">
  <response>
    <statistics domain="biysk.local" from="2019/11/05 00:00:00" to="2019/11/18 00:00:00"
      system="ecss10.ssw" subsystem="trunk_manager" statistic="incoming_seizures.counter"
      type="counter" measure_type="day" sum="1654.000000" variable_part_1="smg-4">
      <value time="2019/11/05 16:59:59" value="79" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/06 16:59:59" value="134" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/07 16:59:59" value="97" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/08 16:59:59" value="163" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/09 16:59:59" value="82" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/10 16:59:59" value="57" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/11 16:59:59" value="140" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/12 16:59:59" value="148" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/13 16:59:59" value="219" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/14 16:59:59" value="155" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/15 16:59:59" value="181" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/16 16:59:59" value="78" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/17 16:59:59" value="121" n="-" min="-" max="-" avg="-"/>
    </statistics>
    <statistics domain="biysk.local" from="2019/11/05 00:00:00" to="2019/11/18 00:00:00"
      system="ecss10.ssw" subsystem="trunk_manager" statistic="incoming_accepted_seizures.counter"
      type="counter" measure_type="day" sum="1654.000000" variable_part_1="smg-4">
      <value time="2019/11/05 16:59:59" value="79" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/06 16:59:59" value="134" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/07 16:59:59" value="97" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/08 16:59:59" value="163" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/09 16:59:59" value="82" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/10 16:59:59" value="57" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/11 16:59:59" value="140" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/12 16:59:59" value="148" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/13 16:59:59" value="219" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/14 16:59:59" value="155" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/15 16:59:59" value="181" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/16 16:59:59" value="78" n="-" min="-" max="-" avg="-"/>
      <value time="2019/11/17 16:59:59" value="121" n="-" min="-" max="-" avg="-"/>
    </statistics>
  </response>
</out>

```

Нс_ecss_statistics_get_metric

Команда предназначена для получения метрик статистики ECSS-10

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ecss_statistics_get_metric

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="statistics" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="system" type="xs:string"/>
                  <xs:attribute name="subsystem" type="xs:string"/>
                  <xs:attribute name="metric" type="xs:string"/>
                  <xs:attribute name="variable_part_1" type="xs:string"
use="optional"/>
                  <xs:attribute name="variable_part_2" type="xs:string"
use="optional"/>
                  <xs:attribute name="type" type="xs:string"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="statistics" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="system" type="xs:string"/>
                  <xs:attribute name="subsystem" type="xs:string"/>
                  <xs:attribute name="metric" type="xs:string"/>
                  <xs:attribute name="variable_part_1" type="xs:string"
use="optional"/>
                  <xs:attribute name="variable_part_2" type="xs:string"
use="optional"/>
                  <xs:attribute name="type" type="xs:string"/>
                  <xs:attribute name="value" type="xs:int"/>
                  <xs:attribute name="time" type="xs:string"/>

```

```

        <xs:attribute name="n" type="xs:int" use="optional"/>
        <xs:attribute name="min" type="xs:int" use="optional"/>
        <xs:attribute name="max" type="xs:int" use="optional"/>
        <xs:attribute name="avg" type="xs:double" use="optional"/>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/ecss_statistics_get_metric

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ecss_statistics_get_metric.xsd">
  <request domain="biysk.local">
    <statistics system="ecss10.ssw" subsystem="core" metric="ecss_statistics_get_metric"
      type="counter"/>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ecss_statistics_get_metric.xsd">
  <response/>
</out>

```

Нс_ftp_cluster_options - Управление настройками FTP подсистемы TTS

- [Нс_ftp_cluster_options_info](#)
- [Нс_ftp_cluster_options_set](#)
- [Нс_ftp_cluster_options_clean](#)

[Нс_ftp_cluster_options_info](#)

Просмотр значений опций FTP подсистемы TTS уровня ноды и кластера.

Аналог команды в CoCon:

cluster/core/<CLUSTER>/tts/ftp/properties/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ftp_cluster_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="ftp_cluster_options_info">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get FTP cluster options.
      Usage: http://server:9999/commands/ftp_cluster_options_info
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <!-- Network port -->
  <xs:simpleType name="portType">
    <xs:restriction base="positiveIntegerType">
      <xs:minExclusive value="1"/>
      <xs:maxInclusive value="49151"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- Options for TTS node -->
  <xs:complexType name="FtpClusterOptions">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="password" type="nonEmptyNormalizedStringType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              In case of strong security restrictions password may be shown as
              *****
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="port" type="portType" use="optional" />
        <xs:attribute name="iface" type="nonEmptyNormalizedStringType" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="nc_options" type="ncInfoInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

```
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:all minOccurs="1" maxOccurs="1">
            <xs:element name="nc_options" type="ncInfoOutType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>
```

Пример 1:

Запрос:

http://192.168.1.21:9999/commands/ftp_cluster_options_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request>
    <nc_options cluster="core1"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ftp_cluster_options_info.xsd">
  <result>
    <nc_options>
      <cluster name="core1">
        <options xs:type="FtpClusterOptions" port="21" password="*****"
iface="0.0.0.0"/>
      </cluster>
    </nc_options>
  </result>
</out>

```

Пример 2:

Запрос:

http://192.168.23.34:9999/commands/ftp_cluster_options_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request>
    <nc_options cluster="core1" node="core1@ecss1"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ftp_cluster_options_info.xsd">
  <result>
    <nc_options>
      <cluster name="core1">
        <options xs:type="FtpClusterOptions"/>
      </cluster>
      <node name="core1@ecss1">
        <options xs:type="FtpClusterOptions" port="21" password="*****"
iface="0.0.0.0"/>
      </node>
    </nc_options>
  </result>
</out>

```


Нс_ftp_cluster_options_set

Установка значения опций FTP подсистемы TTS уровня ноды и кластера.

Аналог команды в CoCon:

```
cluster/core/<CLUSTER>/tts/ftp/properties/set
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ftp_cluster_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" id="ftp_cluster_options_set">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set ftp cluster options.
      Usage: http://server:9999/commands/ftp_cluster_options_set
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <!-- Network port -->
  <xs:simpleType name="portType">
    <xs:restriction base="positiveIntegerType">
      <xs:minExclusive value="1"/>
      <xs:maxInclusive value="49151"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="FtpClusterOptions">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="password" type="nonEmptyNormalizedStringType" use="optional"
          />
        <xs:attribute name="port" type="portType" use="optional" />
        <xs:attribute name="iface" type="nonEmptyNormalizedStringType" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="nc_options" type="ncSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>

```

```

        <xs:all minOccurs="1" maxOccurs="1">
            <xs:element name="nc_options" type="ncSetOutType"/>
        </xs:all>
    </xs:complexType>
</xs:element>
<xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/ftp_cluster_options_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_ftp_cluster_options_set.xsd">
  <request>
    <nc_options>
      <cluster name="core1">
        <options xs:type="FtpClusterOptions" iface="127.0.0.1" password="cdr" port="21"/>
      </cluster>
      <node name="core1@ecss1">
        <options xs:type="FtpClusterOptions" port="10021"/>
      </node>
    </nc_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ftp_cluster_options_set.xsd">
  <result>
    <nc_options>
      <cluster name="core1">
        <success option="iface"/>
        <success option="password"/>
        <success option="port"/>
      </cluster>
      <node name="core1@ecss1">
        <success option="port"/>
      </node>
    </nc_options>
  </result>
</out>
```

hc_ftp_cluster_options_clean

Сброс значений опций FTP подсистемы TTS уровня ноды и кластера.

Аналог команды в CoCon:

cluster/core/<CLUSTER>/tts/ftp/properties/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ftp_cluster_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="ftp_cluster_options_clean">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean ftp cluster options.
      Usage: http://server:9999/commands/ftp\_cluster\_options\_clean
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="nc_options" type="ncCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="nc_options" type="ncCleanOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/ftp_cluster_options_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_ftp_cluster_options_clean.xsd">
  <request>
    <nc_options>
      <cluster name="core1">
        <option name="iface"/>
        <option name="password"/>
        <option name="port"/>
      </cluster>
      <node name="core1@ecss1">
        <option name="port"/>
      </node>
    </nc_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ftp_cluster_options_clean.xsd">
  <result>
    <nc_options>
      <cluster name="core1">
        <success option="iface"/>
        <success option="password"/>
        <success option="port"/>
      </cluster>
      <node name="core1@ecss1">
        <success option="port"/>
      </node>
    </nc_options>
  </result>
</out>

```

Hc_gateway_port - Управление портами шлюза

- [Hc_gateway_port_info](#)
- [Hc_gateway_port_set](#)
- [Hc_gateway_port_clean](#)

Нс_gateway_port_info

Команда предназначена для просмотра информации по определенным портам шлюзов.

Аналог команды в CoCon:

/gateway/port/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_port_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="port" type="xs:string" use="optional"/>
  </xs:complexType>
  <xs:complexType name="gatewayOutType">
    <xs:sequence>
      <xs:element name="property" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="value" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="port" type="xs:string" use="optional"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="gateway" type="gatewayType" minOccurs="0" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="gateway" type="gatewayOutType" minOccurs="0"
maxOccurs="unbounded" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

Запрос: http://192.168.23.34:9999/commands/gateway_port_info

```

<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_port_info.xsd">
  <gateway domain="p.city" name="gw3" port="p/3" />
</in>

```

Ответ:


```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_port_info.xsd">
  <gateway name="gw3" port="p/3">
    <property name="key1" value="val1"/>
    <property name="key2" value="val2"/>
  </gateway>
</out>
```

Пример 2:

Запрос: http://192.168.23.34:9999/commands/gateway_port_info

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_port_info.xsd" >
  <gateway domain="p.city" />
</in>
```

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_port_info.xsd">
  <gateway name="gw1" port="p/9@sorm_megaco"/>
  <gateway name="gw1" port="p/14@sorm_megaco"/>
  <gateway name="gw1" port="p/13@sorm_megaco"/>
  <gateway name="gw3" port="p/4">
    <property name="key1" value="val1"/>
    <property name="key2" value="val2"/>
  </gateway>
  <gateway name="gw1" port="p/4@sorm_megaco"/>
  <gateway name="gw3" port="p/2"/>
  <gateway name="gw1" port="p/8@sorm_megaco"/>
  <gateway name="gw1" port="p/2@sorm_megaco"/>
  <gateway name="gw1" port="p/15@sorm_megaco"/>
  <gateway name="gw1" port="p/16@sorm_megaco"/>
  <gateway name="gw1" port="p/12@sorm_megaco"/>
  <gateway name="gw1" port="p/11@sorm_megaco"/>
  <gateway name="gw1" port="p/7@sorm_megaco"/>
  <gateway name="gw1" port="p/1@sorm_megaco"/>
  <gateway name="gw1" port="p/5@sorm_megaco"/>
  <gateway name="gw1" port="p/3@sorm_megaco"/>
  <gateway name="gw3" port="p/3">
    <property name="key1" value="val1"/>
    <property name="key2" value="val2"/>
  </gateway>
  <gateway name="gw1" port="p/10@sorm_megaco"/>
  <gateway name="gw1" port="p/6@sorm_megaco"/>
  <gateway name="gw1" port="p/0@sorm_megaco"/>
</out>

```

hc_gateway_port_set

Команда предназначена для установки свойств портов шлюзов.

Аналог команды в CoCon:

/gateway/port/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_port_set

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:sequence>
      <xs:element name="property" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="value" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="port" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="gateway" type="gatewayType" minOccurs="1" maxOccurs="1" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_port_set

```

<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="gateway_port_set.xsd">
  <gateway domain="p.city" name="gw2" port="p/1">
    <property name="k1" value="12223" />
    <property name="k2" value="[1, 4, 3]" />
  </gateway>
</in>

```

Нс_gateway_port_clean

Команда предназначена для удаления свойств портов шлюзов.

Аналог команды в CoCon:

/gateway/port/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_port_clean

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:sequence>
      <xs:element name="property" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="port" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="gateway" type="gatewayType" minOccurs="1" />
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_port_clean

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="../priv/
xsd/gateway_port_clean.xsd">
  <gateway domain="p.city" name="gw2" port="p/1">
    <property name="k1"/>
  </gateway>
</in>
```

Hc_gateway_profile - Управление профилями настроек шлюза

- [Hc_gateway_profile_declare](#)
- [Hc_gateway_profile_info](#)
- [Hc_gateway_profile_set](#)
- [Hc_gateway_profile_clean](#)
- [Hc_gateway_profile_remove](#)

Hc_gateway_profile_declare

Команда предназначена для создания нового профиля шлюза.

Аналог команды в CoCon:

`/gateway/profile/declare`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/gateway_profile_declare`

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="profileType">
    <xs:sequence>
      <xs:element name="property" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="value" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="profile" type="profileType" minOccurs="1" />
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_profile_declare

```

<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="gateway_profile_declare.xsd">
  <profile domain="p.city" name="profile5">
    <property name="key1" value="abc" />
    <property name="key2" value="[1, 2, 3, 5]" />
  </profile>
</in>

```

Нс_gateway_profile_info

Команда предназначена для просмотра информации по имеющимся профилям шлюзов.

Аналог команды в CoCon:

/gateway/profile/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_profile_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="profileType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="optional"/>
  </xs:complexType>
  <xs:complexType name="profileOutType">
    <xs:sequence>
      <xs:element name="property" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="value" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="profile" type="profileType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="profile" type="profileOutType" minOccurs="0"
maxOccurs="unbounded" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример 1:

Запрос: http://192.168.23.34:9999/commands/gateway_profile_info

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_profile_info.xsd">
  <profile domain="p.city" name="profile5" />
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_profile_info.xsd">
  <profile name="profile5">
    <property name="key1" value="abc"/>
    <property name="key2" value="[1,2,3,5]"/>
  </profile>
</out>
```

Пример 2:

Запрос: http://192.168.23.34:9999/commands/gateway_profile_info

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_profile_info.xsd" >
  <profile domain="p.city" />
</in>
```

Ответ:


```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_profile_info.xsd">
  <profile name="profile6">
    <property name="key1" value="abc"/>
    <property name="key2" value="[1,2,3,5]"/>
  </profile>
  <profile name="profile5">
    <property name="key1" value="abc"/>
    <property name="key2" value="[1,2,3,5]"/>
  </profile>
  <profile name="profile4">
    <property name="k1" value="12223"/>
    <property name="k2" value="[1,4,3]"/>
    <property name="k3" value="123"/>
  </profile>
  <profile name="profile3"/>
  <profile name="profile2">
    <property name="key1" value="value1"/>
    <property name="key2" value="[1,2,3]"/>
  </profile>
  <profile name="profile1"/>
  <profile name="profile10"/>
</out>
```

Нс_gateway_profile_set

Команда предназначена для установки свойств для определенного профиля шлюза.

Аналог команды в CoCon:

/gateway/profile/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_profile_set

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="profileType">
    <xs:sequence>
      <xs:element name="property" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="value" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="profile" type="profileType" minOccurs="1" />
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_profile_set

```

<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_profile_set.xsd">
  <profile domain="p.city" name="profile4">
    <property name="k1" value="123" />
    <property name="k2" value="[1, 2, 3]" />
  </profile>
</in>

```

Нс_gateway_profile_clean

Команда предназначена для удаления свойств определенного профиля шлюза.

Аналог команды в CoCon:

/gateway/profile/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_profile_clean

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="profileType">
    <xs:sequence>
      <xs:element name="property" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="profile" type="profileType" minOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_profile_clean

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="gateway_profile_clean.xsd">
  <profile domain="p.city" name="profile1">
    <property name="key1" />
    <property name="key2" />
  </profile>
</in>
```

Нс_gateway_profile_remove

Команда предназначена для удаления профиля шлюза из виртуальной АТС.

Аналог команды в CoCon:**gateway/profile/remove****Метод HTTP-запроса:**

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_profile_remove

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd" />

  <xs:complexType name="profileType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="profile" type="profileType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="error" type="errorType" />
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример 1:

Запрос: http://192.168.23.34:9999/commands/gateway_profile_remove

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_profile_remove.xsd">
  <profile domain="p.city" name="profile5" />
</in>
```

Ответ:

204

Hc_gateway_sdp - Управление SDP-шаблонами

- [Hc_gateway_sdp_template_declare](#)
- [Hc_gateway_sdp_template_remove](#)
- [Hc_gateway_sdp_template_info](#)
- [Hc_gateway_sdp_template_set](#)

Hc_gateway_sdp_template_declare

Команда предназначена для создания нового шаблона SDP.

Аналог команды в CoCon:

/gateway/sdp-template/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_sdp_template_declare

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name = "nameType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="v"/>
      <xs:enumeration value="o"/>
      <xs:enumeration value="s"/>
      <xs:enumeration value="i"/>
      <xs:enumeration value="u"/>
      <xs:enumeration value="e"/>
      <xs:enumeration value="p"/>
      <xs:enumeration value="c"/>
      <xs:enumeration value="b"/>
      <xs:enumeration value="z"/>
      <xs:enumeration value="k"/>
      <xs:enumeration value="a"/>
      <xs:enumeration value="t"/>
      <xs:enumeration value="r"/>
      <xs:enumeration value="m"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="sdp-templateType">
    <xs:sequence>
      <xs:element name="property" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="nameType" use="required"/>
          <xs:attribute name="value" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="sdp-template" type="sdp-templateType" minOccurs="1" />
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_sdp_template_declare

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_sdp_template_declare.xsd">
  <sdp-template domain="p.city" name="template4">
    <property name="a" value="123" />
    <property name="b" value="[1, 2, 3]" />
  </sdp-template>
</in>
```

hc_gateway_sdp_template_remove

Команда предназначена для удаления шаблона SDP.

Аналог команды в CoCon:

/gateway/sdp-template/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_sdp_template_remove

Код HTTP-ответа:

- 300 - в случае модала;
- 204 - в случае успеха;
- 404 - в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="templateType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="template" type="templateType" minOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_sdp_template_remove

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_sdp_template_remove.xsd">
  <template domain="p.city" name="template1" />
</in>
```

Может вернуть модал:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="modal.xsd">
  <modal ref="{md@sky, {1315,393114,432462}}" from="remove"
    header="SDP template "template1" is used in gateway(s) ["gw1"]. Before remove SDP
template the system remove it from the gateways."
    passwd="false" results="yesno" error="You should enter 'yes' or 'no'." prompt="Are you
sure"/>
</out>
```

Ис_ gateway_sdp_template_info

Команда предназначена для просмотра информации по имеющимся шаблонам SDP.

Аналог команды в CoCon:

/gateway/sdp-template/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_sdp_template_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="sdp-templateType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="optional"/>
  </xs:complexType>
  <xs:complexType name="sdp-templateOutType">
    <xs:sequence>
      <xs:element name="property" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="value" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="returnType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="sdp-template" type="sdp-templateType" minOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="sdp-template" type="sdp-templateOutType" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

Запрос: http://192.168.23.34:9999/commands/gateway_sdp_template_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="../priv/
xsd/gateway_sdp_template_info.xsd">
  <sdp-template domain="p.city" name="template1" />
</in>

```

Ответ:

```
<?xml version="1.0"?><out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="gateway_sdp_template_info.xsd">
  <sdp-template name="template1">
    <property name="c" value="123"/>
    <property name="t" value="1234"/>
  </sdp-template>
</out>
```

Пример 2:

Запрос: http://192.168.23.34:9999/commands/gateway_sdp_template_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="..priv/
xsd/gateway_sdp_template_info.xsd">
  <sdp-template domain="p.city" />
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="gateway_sdp_template_info.xsd">
  <sdp-template name="template4">
    <property name="b" value="434"/>
  </sdp-template>
  <sdp-template name="template1">
    <property name="c" value="123"/>
    <property name="t" value="1234"/>
  </sdp-template>
  <sdp-template name="template3"/>
</out>
```

Ис_ gateway_sdp_template_set

Команда предназначена для установки свойств определенного шаблона SDP.

Аналог команды в CoCon:

/gateway/sdp-template/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_sdp_template_set

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd" />

  <xs:simpleType name = "nameType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="v"/>
      <xs:enumeration value="o"/>
      <xs:enumeration value="s"/>
      <xs:enumeration value="i"/>
      <xs:enumeration value="u"/>
      <xs:enumeration value="e"/>
      <xs:enumeration value="p"/>
      <xs:enumeration value="c"/>
      <xs:enumeration value="b"/>
      <xs:enumeration value="z"/>
      <xs:enumeration value="k"/>
      <xs:enumeration value="a"/>
      <xs:enumeration value="t"/>
      <xs:enumeration value="r"/>
      <xs:enumeration value="m"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="sdp-templateType">
    <xs:sequence>
      <xs:element name="property" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="nameType" use="required"/>
          <xs:attribute name="value" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="sdp-template" type="sdp-templateType" minOccurs="1" />
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="error" type="errorType" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_sdp_template_set

```

<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_sdp_template_set.xsd">
  <sdp-template domain="p.city" name="profile4">
    <property name="a" value="123" />
    <property name="b" value="[1, 2, 3]" />
  </sdp-template>
</in>

```

Hc_gateway - Управление шлюзами

- [Hc_gateway_declare](#)
- [Hc_gateway_info](#)
- [Hc_gateway_remove](#)
- [Hc_gateway_set](#)
- [Hc_gateway_clean](#)
- [Hc_gateway_add_port](#)
- [Hc_gateway_remove_port](#)
- [Hc_gateway_add_sdp_template](#)
- [Hc_gateway_remove_sdp_template](#)
- [Hc_gateway_change_profile](#)

Hc_gateway_declare

Команда создает новый шлюз с указанием типа и профиля.

Аналог команды в CoCon:

/gateway/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_declare

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="typeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="megaco"/>
      <xs:enumeration value="sip"/>
      <xs:enumeration value="sigtran_m2ua"/>
      <xs:enumeration value="sigtran_m3ua"/>
      <xs:enumeration value="sigtran_iua"/>
      <xs:enumeration value="sigtran_trunk"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="profileType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="typeType" use="required"/>
    <xs:attribute name="profile" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="gateway" type="profileType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_declare

Параметры:

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="gateway_declare.xsd">
  <gateway domain="p.city" name="fdf" type="sip" profile="profile1" />
</in>
```

Нс_gateway_info

Команда предназначена для просмотра информации о шлюзе(шлюзах).

Аналог команды в CoCon:

/gateway/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- gatewayInType -->
  <xs:complexType name="gatewayInType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="optional"/>
    <xs:attribute name="type" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- portType -->
  <xs:complexType name="portType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- sdpTemplateType -->
  <xs:complexType name="sdpTemplateType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- propertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- gatewayOutType -->
  <xs:complexType name="gatewayOutType">
    <xs:all>
      <xs:element name="ports">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="port" type="portType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="templates">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="sdp-template" type="sdpTemplateType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="properties">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>

```



```

    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="xs:string" use="required"/>
    <xs:attribute name="profile" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="gateway" type="gatewayInType" minOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="gateway" type="gatewayOutType" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

Запрос: http://192.168.23.34:9999/commands/gateway_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_info.xsd">
  <gateway domain="p.city" name="gw3" />
</in>

```

Ответ:

```
<?xml version="1.0"?>
  <out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="gateway_info.xsd">
    <gateway domain="p.city" name="gw3" profile="profile2" type="sip">
      <ports>
        <port name="p/4"/>
        <port name="p/5"/>
        <port name="p/6"/>
      </ports>
      <properties>
        <property name="k6" value="6"/>
        <property name="k4" value="34"/>
        <property name="k1" value="123"/>
        <property name="k2" value="[1,2,3]"/>
      </properties>
    </gateway>
  </out>
```

Пример 2:

Запрос:

http://192.168.23.34:9999/commands/gateway_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="gateway_info.xsd">
  <gateway domain="p.city" type="megaco" />
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="gateway_info.xsd">
  <gateway domain="p.city" name="gw2" type="megaco" profile="tgw">
    <ports/>
    <templates/>
    <properties>
      <property name="fax_detect" value="caller"/>
      <property name="location" value="local_network"/>
    </properties>
  </gateway>
</out>
```

Нс_gateway_remove

Команда предназначена для удаления шлюза.

Аналог команды в CoCon:

/gateway/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_remove

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="gateway" type="gatewayType" minOccurs="1" maxOccurs="unbounded"
/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_remove

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_remove.xsd">
  <gateway domain="p.city" name="gw2" />
  <gateway domain="p.city" name="gw3" />
</in>
```

Нс_gateway_set

Команда предназначена для установки свойств определенного шлюза.

Аналог команды в CoCon:

/gateway/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_set

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:sequence>
      <xs:element name="property" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="value" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="gateway" type="gatewayType" minOccurs="1" />
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_set

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_set.xsd">
  <gateway domain="p.city" name="gw3">
    <property name="k1" value="123" />
    <property name="k2" value="[1, 2, 3]" />
  </gateway>
</in>
```

Нс_gateway_clean

Команда предназначена для удаления свойств определенного шлюза.

Аналог команды в CoCon:

/gateway/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_clean

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:sequence>
      <xs:element name="property" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="gateway" type="gatewayType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_clean

```

<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_clean.xsd">
  <gateway domain="p.city" name="gw3">
    <property name="key1" />
    <property name="key2" />
  </gateway>
</in>

```

Нс_gateway_add_port

Команда предназначена для добавления нового порта к шлюзу.

Аналог команды в CoCon:

/gateway/add-port

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_add_port

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:sequence>
      <xs:element name="property" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="value" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="port" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="gateway" type="gatewayType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_add_port

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="gateway_add_port.xsd">
  <gateway domain="p.city" name="gw3" port="p/3">
    <property name="key1" value="val1" />
    <property name="key2" value="val2" />
  </gateway>
</in>
```

Нс_gateway_remove_port

Команда предназначена для удаления порта из шлюза.

Аналог команды в CoCon:

/gateway/remove-port**Метод HTTP-запроса:**

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_remove_port

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="port" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="gateway" type="gatewayType" minOccurs="1" maxOccurs="unbounded"
        />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_remove_port

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="gateway_remove_port.xsd">
  <gateway domain="p.city" name="gw2" port="p/1"/>
  <gateway domain="p.city" name="gw2" port="p/2"/>
</in>
```

hc_gateway_add_sdp_template

Команда предназначена для добавления шаблона SDP к шлюзу.

Аналог команды в CoSop:

/gateway/add-sdp-template**Метод HTTP-запроса:**

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_add_sdp_template

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:sequence>
      <xs:element name="sdp-template" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="gateway" type="gatewayType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_add_sdp_template

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="gateway_add_sdp_template.xsd">
  <gateway domain="p.city" name="gw3">
    <sdp-template name="template3" />
    <sdp-template name="template2" />
  </gateway>
</in>
```

hc_gateway_remove_sdp_template

Команда предназначена для удаления шаблона SDP из шлюза.

Аналог команды в CoCon:

`/gateway/remove-sdp-template`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/gateway_remove_sdp_template`

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:sequence>
      <xs:element name="sdp-template" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="gateway" type="gatewayType" minOccurs="1" maxOccurs="1" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: `http://192.168.23.34:9999/commands/gateway_remove_sdp_template`

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="../priv/
xsd/gateway_remove_sdp_template.xsd">
  <gateway domain="p.city" name="gw3">
    <sdp-template name="template3" />
    <sdp-template name="template2" />
  </gateway>
</in>
```

Нс_gateway_change_profile

Команда предназначена для изменения профиля шлюза.

Аналог команды в CoCon:

/gateway/change-profile

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/gateway_change_profile

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:sequence>
      <xs:element name="profile">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="gateway" type="gatewayType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/gateway_change_profile

```

<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="../priv/
xsd/gateway_change_profile.xsd">
  <gateway domain="p.city" name="gw3">
    <profile name="profile3" />
  </gateway>
</in>

```

Нс_get_extended_subscriber_card - Вывод информационной карточки абонента

[Нс_get_extended_subscriber_card](#)

Команда предназначена для вывода расширенной информационной карточки абонента.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/get_extended_subscriber_card

Код HTTP-ответа:

- 200 – в случае успеха.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- sipRegistrationType-->
  <xs:complexType name="sipRegistrationType">
    <xs:attribute name="access_point" type="xs:string" use="required"/>
    <xs:attribute name="reregister_date" type="xs:string" use="required"/>
    <xs:attribute name="ttl" type="xs:integer" use="required"/>
    <xs:attribute name="expires" type="xs:integer" use="required"/>
    <xs:attribute name="uri" type="xs:string" use="required"/>
    <xs:attribute name="first_via_host" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- sipRegistrationsType-->
  <xs:complexType name="sipRegistrationsType">
    <xs:sequence>
      <xs:element name="registration" type="sipRegistrationType" minOccurs="0" maxOccurs="u
nbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- callType-->
  <xs:complexType name="callType">
    <xs:attribute name="from" type="xs:string" use="required"/>
    <xs:attribute name="to" type="xs:string" use="required"/>
    <xs:attribute name="duration" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- activeCallsType-->
  <xs:complexType name="activeCallsType">
    <xs:sequence>
      <xs:element name="call" type="callType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="regimeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="private_in" type="xs:boolean" use="required"/>
    <xs:attribute name="private_out" type="xs:boolean" use="required"/>
    <xs:attribute name="local_in" type="xs:boolean" use="required"/>
    <xs:attribute name="local_out" type="xs:boolean" use="required"/>
    <xs:attribute name="zone_in" type="xs:boolean" use="required"/>
    <xs:attribute name="zone_out" type="xs:boolean" use="required"/>
    <xs:attribute name="intercity_in" type="xs:boolean" use="required"/>
    <xs:attribute name="intercity_out" type="xs:boolean" use="required"/>
    <xs:attribute name="international_in" type="xs:boolean" use="required"/>
    <xs:attribute name="international_out" type="xs:boolean" use="required"/>
    <xs:attribute name="emergency_in" type="xs:boolean" use="required"/>
    <xs:attribute name="emergency_out" type="xs:boolean" use="required"/>
  </xs:complexType>
  <!-- propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>

```

```

    <xs:attribute name="value" type="xs:string" use="required"/>
</xs:complexType>
<!-- ssType-->
<xs:complexType name="ssType">
  <xs:sequence>
    <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="family" type="xs:string" use="optional">
    <xs:annotation>
      <xs:documentation xml:lang="en">
        Obsolete attribute. Use name instead of It.
      </xs:documentation>
      <xs:documentation xml:lang="ru">
        Устаревший атрибут. Теперь ДВО полностью определяется по атрибуту name.
      </xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
<!-- sssType-->
<xs:complexType name="sssType">
  <xs:sequence>
    <xs:element name="ss" type="ssType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!-- viaType-->
<xs:complexType name="viaType">
  <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>
<!-- viaListType-->
<xs:complexType name="viaListType">
  <xs:sequence>
    <xs:element name="via" type="viaType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!-- aliasType-->
<xs:complexType name="aliasType">
  <xs:all>
    <xs:element name="regime" type="regimeType"/>
    <xs:element name="access_type" type="regimeType"/>
    <xs:element name="via_list" type="viaListType"/>
    <xs:element name="sss" type="sssType"/>
    <xs:element name="registrations" type="sipRegistrationsType"/>
    <xs:element name="active_calls" type="activeCallsType"/>
  </xs:all>
  <xs:attribute name="domain" type="xs:string" use="required"/>
  <xs:attribute name="iface" type="xs:string" use="required"/>
  <xs:attribute name="address" type="xs:string" use="required"/>
  <xs:attribute name="active" type="xs:boolean" use="required"/>
  <xs:attribute name="accessable_in" type="xs:boolean" use="required"/>

```

```

<xs:attribute name="last_registration_time" type="xs:string" use="required"/>
<xs:attribute name="accessibility_check_result" type="xs:string" use="required"/>
<xs:attribute name="category" type="xs:string" use="required"/>
<xs:attribute name="login" type="xs:string" use="required"/>
<xs:attribute name="password" type="xs:string" use="required">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      In case of strong security restrictions password may be shown as *****
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
</xs:complexType>
<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:attribute name="address" type="xs:string" use="required"/>
          <xs:attribute name="domain" type="xs:string" use="required"/>
          <xs:attribute name="timerF" use="optional">
            <xs:simpleType>
              <xs:restriction base="xs:integer">
                <xs:minInclusive value="500"/>
                <xs:maxInclusive value="20000"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
          <xs:attribute name="sip_cluster" type="xs:string" use="optional">
            <xs:annotation>
              <xs:documentation xml:lang="en">
                Obsolete attribute.
              </xs:documentation>
            </xs:annotation>
          </xs:attribute>
          <xs:attribute name="core_cluster" type="xs:string" use="optional">
            <xs:annotation>
              <xs:documentation xml:lang="en">
                Obsolete attribute.
              </xs:documentation>
            </xs:annotation>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>

```



```

    <xs:all>
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="alias" type="aliasType" minOccurs="0" maxOccurs="un
bounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/get_extended_subscriber_card

```

<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request address="240101" domain="biysk.local"/>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="get_extended_subscriber_card.xsd">
  <response>
    <alias domain="biysk.local" iface="064bc9652bfe3487" address="240101" active="true"
    accessible_in="true" category="ordinarySubscriber" login="240101" password="ext_101_SSW-3"
    last_registration_time="11.03.2021 09:30:42" accessibility_check_result="accessible: true,
    OPTIONS to <sip:240101@192.168.2.26>: 200 OK, cseq: 347119 OPTIONS. Request by node: sip1@ecss1">
    <regime name="" private_in="true" private_out="true" local_in="true" local_out="true"
    zone_in="true" zone_out="true" intercity_in="true" intercity_out="true" international_in="true"
    international_out="true" emergency_in="true" emergency_out="true"/>
    <access_type name="" private_in="true" private_out="true" local_in="true" local_out="
    true" zone_in="true" zone_out="true" intercity_in="true" intercity_out="true" international_in="t
    rue" international_out="true" emergency_in="true" emergency_out="true"/>
    <via_list>
      <via name="SIP/2.0/UDP 192.168.2.26:5060"/>
    </via_list>
    <sss>
      <ss name="cc_agent" family="CCAgent">
        <property key="extra_number" value="[]"/>
      </ss>
      <ss name="chold" family="CHOLD">
        <property key="disable_moh" value="false"/>
        <property key="dtmf_sequence_as_flash" value="false"/>
      </ss>
      <ss name="clip" family="CLIP"/>
      <ss name="cnip" family="CNIP"/>
      <ss name="conference" family="Conference">
        <property key="destroy_mode" value="by_no_master"/>
        <property key="max_participants" value="16"/>
      </ss>
      <ss name="ctr" family="CTR"/>
      <ss name="meet_me_conference" family="meet_me_conference">
        <property key="mode" value="master"/>
        <property key="destroy_mode" value="by_no_more_calls"/>
        <property key="room_number" value="[]"/>
      </ss>
      <ss name="teleconference" family="teleconference"/>
      <ss name="teleconference_manager" family="Teleconference">
        <property key="second_line" value="[]"/>
      </ss>
    </sss>
    <registrations>
      <registration access_point="192.168.2.26:5060/UDP" reregister_date="11.03.2021
    09:30:42" ttl="900" expires="546" uri="sip:240101@192.168.2.26:5060;transport=UDP;line=20457"
    first_via_host="192.168.2.26"/>
    </registrations>
    <active_calls/>
  </response>
</out>

```

```
</alias>  
</response>  
</out>
```

Нс_get_info - Время работы узла(ов) с момента последнего перезапуска

Нс_get_info

Команда возвращает время работы узла(ов) с момента последнего перезапуска.

Аналог команды в CoCon:

`/node/<NODE_NAME>/uptime`

`/node/uptime`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/get_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="releaseType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="version" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="uptimeType">
    <xs:sequence>
      <xs:element name="release" type="releaseType" minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="days" type="xs:string" use="required"/>
    <xs:attribute name="hours" type="xs:string" use="required"/>
    <xs:attribute name="minutes" type="xs:string" use="required"/>
    <xs:attribute name="seconds" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="node" type="nodeType" minOccurs="0" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="uptime" type="uptimeType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Информация о конкретной подсистеме:

Запрос: http://192.168.1.21:9999/commands/get_info

```

<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="get_info.x
sd">
  <node name="md1@ecss1" />
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="get_info.x
sd">
  <uptime name="md1@ecss1" days="0" hours="4" minutes="47" seconds="14">
    <release name="ecss-mediator" version="3.14.3.43"/>
  </uptime>
</out>
```

Информация о всех подсистемах:

Запрос: http://192.168.1.21:9999/commands/get_info

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="get_info.x
sd" />
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="get_info.x
sd">
  <uptime name="sip1@ecss1" days="0" hours="4" minutes="48" seconds="5">
    <release name="ecss-pa-sip" version="3.14.3.43"/>
  </uptime>
  <uptime name="mycelium1@ecss1" days="0" hours="4" minutes="48" seconds="50">
    <release name="ecss-mycelium" version="3.14.3.43"/>
  </uptime>
  <uptime name="megaco1@ecss1" days="0" hours="4" minutes="47" seconds="47">
    <release name="ecss-pa-megaco" version="3.14.3.43"/>
  </uptime>
  <uptime name="md1@ecss1" days="0" hours="4" minutes="48" seconds="4">
    <release name="ecss-mediator" version="3.14.3.43"/>
  </uptime>
  <uptime name="ds1@ecss1" days="0" hours="4" minutes="48" seconds="33">
    <release name="ecss-ds" version="3.14.3.43"/>
  </uptime>
  <uptime name="core1@ecss1" days="0" hours="4" minutes="48" seconds="33">
    <release name="ecss-core" version="3.14.3.43"/>
  </uptime>
</out>
```

Нс_hosts_info - Информация о состоянии серверов

hosts_info

Команда возвращает информацию о состоянии ресурсов серверов - CPU, память, дисковая подсистема, сетевые интерфейсы.

Аналог команды в CoCon:

node/hosts-info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/hosts_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- HostInfoType-->
  <xs:complexType name="HostInfoType">
    <xs:all>
      <xs:element name="cpu" type="ItemsListType"/>
      <xs:element name="memory" type="ItemsListType"/>
      <xs:element name="memoryTotal" type="ItemsListType"/>
      <xs:element name="disk" type="DiskInfoType"/>
      <xs:element name="iface" type="IfaceInfoType"/>
      <xs:element name="node" type="NodeInfoType"/>
    </xs:all>
    <xs:attribute name="node" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- ItemsListType-->
  <xs:complexType name="ItemsListType">
    <xs:sequence>
      <xs:element name="item" type="ItemType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- ItemType-->
  <xs:complexType name="ItemType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- DiskInfoType-->
  <xs:complexType name="DiskInfoType">
    <xs:sequence>
      <xs:element name="item" type="DiskItemType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- DiskItemType-->
  <xs:complexType name="DiskItemType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
    <xs:attribute name="used" type="xs:string" use="required"/> <!-- 0-100% usage -->
  </xs:complexType>
  <!-- IfaceInfoType-->
  <xs:complexType name="IfaceInfoType">
    <xs:sequence>
      <xs:element name="item" type="IfaceItemType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- IfaceItemType-->
  <xs:complexType name="IfaceItemType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="macaddr" type="xs:string" use="required"/>
    <xs:attribute name="addr" type="xs:string" use="required"/>
    <xs:attribute name="netmask" type="xs:string" use="required"/>
  </xs:complexType>

```

```
    <xs:attribute name="dstaddr" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- NodeInfoType -->
  <xs:complexType name="NodeInfoType">
    <xs:sequence>
      <xs:element name="item" type="NodeItemType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- NodeItemType -->
  <xs:complexType name="NodeItemType">
    <xs:attribute name="node" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="host" type="HostInfoType" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример

Запрос: http://192.168.1.21:9999/commands/hosts_info

```
<in/>
```

Ответ:200


```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_hosts_info.xsd">
  <host node="core1@ecss1">
    <cpu>
      <item name="Total" value="20.59"/>
      <item name="CPU #1" value="18.75"/>
      <item name="CPU #2" value="11.76"/>
    </cpu>
    <memory>
      <item name="total" value="372827688"/>
      <item name="processes" value="130079880"/>
      <item name="processes_used" value="130078488"/>
      <item name="system" value="242747808"/>
      <item name="atom" value="3223825"/>
      <item name="atom_used" value="3211956"/>
      <item name="binary" value="31969736"/>
      <item name="code" value="71185090"/>
      <item name="ets" value="30218280"/>
    </memory>
    <memoryTotal>
      <item name="system_total_memory" value="4096815104"/>
      <item name="free_swap" value="0"/>
      <item name="total_swap" value="0"/>
      <item name="cached_memory" value="342142976"/>
      <item name="buffered_memory" value="168120320"/>
      <item name="free_memory" value="214089728"/>
      <item name="total_memory" value="4096815104"/>
    </memoryTotal>
    <disk>
      <item name="/" value="52576092160" used="26"/>
      <item name="/dev" value="2012602368" used="0"/>
      <item name="/dev/shm" value="2048405504" used="0"/>
      <item name="/run" value="409681920" used="1"/>
      <item name="/run/lock" value="5242880" used="0"/>
      <item name="/sys/fs/cgroup" value="2048405504" used="0"/>
      <item name="/snap/stress-ng/924" value="3407872" used="100"/>
      <item name="/snap/core/7917" value="93454336" used="100"/>
      <item name="/snap/keepalived/514" value="18743296" used="100"/>
      <item name="/snap/keepalived/618" value="18874368" used="100"/>
      <item name="/var/log" value="21003583488" used="13"/>
      <item name="/var/log/ecss" value="31572529152" used="6"/>
      <item name="/var/lib/ecss" value="31572529152" used="2"/>
      <item name="/var/lib/ecss/glusterfs" value="31572529152" used="1"/>
      <item name="/var/lib/ecss-mysql" value="52576092160" used="2"/>
      <item name="/snap/stress-ng/1046" value="3407872" used="100"/>
      <item name="/snap/core/8039" value="93454336" used="100"/>
    </disk>
  </host>
</out>

```

```

    <iface>
      <item name="lo" macaddr="00:00:00:00:00:00" addr="127.0.0.1" netmask="255.0.0.0"
dstaddr="-"/>
      <item name="enp1s0" macaddr="82:17:A8:3B:DC:FD" addr="-" netmask="-" dstaddr="-"/>
      <item name="enp2s0" macaddr="82:17:A8:3B:DC:FD" addr="-" netmask="-" dstaddr="-"/>
      <item name="bond1" macaddr="82:17:A8:3B:DC:FD" addr="fe80::8017:a8ff:fe3b:dcfd"
netmask="ffff:ffff:ffff:ffff::" dstaddr="-"/>
      <item name="bond1.3" macaddr="82:17:A8:3B:DC:FD" addr="192.168.1.21"
netmask="255.255.255.0" dstaddr="-"/>
      <item name="bond1.476" macaddr="82:17:A8:3B:DC:FD" addr="10.16.33.5"
netmask="255.255.255.0" dstaddr="-"/>
      <item name="bond1.2" macaddr="82:17:A8:3B:DC:FD" addr="192.168.2.21"
netmask="255.255.255.0" dstaddr="-"/>
      <item name="bond1.3:MYSQL" macaddr="" addr="192.168.1.71" netmask="255.255.255.0"
dstaddr="-"/>
      <item name="bond1.2:SIP62" macaddr="" addr="192.168.2.62" netmask="255.255.255.0"
dstaddr="-"/>
      <item name="bond1.2:SIP61" macaddr="" addr="192.168.2.61" netmask="255.255.255.0"
dstaddr="-"/>
    </iface>
    <node>
      <item node="megaco1@ecss1"/>
      <item node="sip1@ecss1"/>
      <item node="md1@ecss1"/>
      <item node="core1@ecss1"/>
      <item node="ds1@ecss1"/>
      <item node="mycelium1@ecss1"/>
      <item node="subscriber-portal@ecss1"/>
    </node>
  </host>
</out>

```

Нс_http_routing_server - Команды управления HTTP серверами маршрутизации

- [Нс_http_routing_servers_change](#)
- [Нс_http_routing_servers_declare](#)
- [Нс_http_routing_servers_info](#)
- [Нс_http_routing_servers_remove](#)
- [Нс_http_routing_servers_set_all](#)

Нс_http_routing_servers_change

Команда изменяет параметры сервера HTTP маршрутизации вызовов.

Аналог команды в CoCon:

/domain/<DOMAIN>/routing/http/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/http_routing_servers_change

Код HTTP-ответа:

200 – в случае успеха;

404 – если сервер с таким идентификатором не существует

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="HttpServer">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="url" type="xs:string" use="required"/>
    <xs:attribute name="api_key" type="xs:string" use="optional"/>
  </xs:complexType>
  <!--in-->
  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="server" type="HttpServer"/>
            </xs:all>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name = "out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:all>
              <xs:element name="ok"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <!--bad. command failed:-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/http_routing_servers_change

Или с помощью curl:

```
curl -X POST 192.168.1.21:9999/commands/http_routing_servers_change -b "token=<Token>";" --data
"<in><request domain="biysk.local"><server id="serv2" url="10.16.33.5"></server></request></in>"
```

```
<in>
  <request domain="biysk.local">
    <server id="serv2" url="10.16.33.5"/>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
  <out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="http_routing_servers_change.xsd">
    <response>
      <ok/>
    </response>
  </out>
```

[Hc_http_routing_servers_declare](#)

Команда добавляет новый сервер HTTP маршрутизации вызовов.

Аналог команды в CoCon:

```
/domain/<DOMAIN>/routing/http/declare
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/http_routing_servers_declare

Код HTTP-ответа:

200 – в случае успеха;

409 – если сервер с данным идентификатором уже существует.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="HttpServer">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="url" type="xs:string" use="required"/>
    <xs:attribute name="api_key" type="xs:string" use="optional"/>
  </xs:complexType>
  <!--in-->
  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="server" type="HttpServer"/>
            </xs:all>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name = "out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:all>
              <xs:element name="ok"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <!--bad. command failed:-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/http_routing_servers_declare

```
<in>
  <request domain="biysk.local">
    <server id="server1" url="192.168.1.21" api_key="151235215wehfhgdgv"/>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="http_routing_servers_declare.xsd">
  <response>
    <ok/>
  </response>
</out>
```

[Hc_http_routing_servers_info](#)

Команда просмотра информации о настроенных серверах HTTP маршрутизации вызовов.

Аналог команды в CoCon:

`/domain/<DOMAIN>/routing/http/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/http_routing_servers_info

Код HTTP-ответа:

200 – в случае успеха;

404 – если сервера с таким идентификатором не существует;

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="HttpServer">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="url" type="xs:string" use="required"/>
    <xs:attribute name="api_key" type="xs:string" use="optional"/>
  </xs:complexType>
  <!--in-->
  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="id" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name = "out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="server" type="HttpServer"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <!--bad. command failed-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример 1:

Запрос:

http://192.168.1.21:9999/commands/http_routing_servers_info


```
<in>
  <request domain="biysk.local"/>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="http_routing_servers_info.xsd">
  <response>
    <servers>
      <server id="server4" url="http://localhost:8788" api_key="meeli5eik1ainahl"/>
      <server id="server3" url="192.168.2.21" api_key="oi6ohcheiw6et2ai"/>
      <server id="server2" url="10.16.33.5" api_key="eeyaang4zo4jaemi"/>
      <server id="server1" url="192.168.1.21" api_key="151235215wehfhgdv"/>
    </servers>
  </response>
</out>
```

Пример 2:

Запрос:

http://192.168.1.21:9999/commands/http_routing_servers_info

```
<in>
  <request domain="biysk.local" id="server2"/>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="http_routing_servers_info.xsd">
  <response>
    <servers>
      <server id="server2" url="10.16.33.5" api_key="eeyaang4zo4jaemi"/>
    </servers>
  </response>
</out>
```

Нс_http_routing_servers_remove

Команда удаляет сервер HTTP маршрутизации вызовов

Аналог команды в CoCon:

`/domain/<DOMAIN>/routing/http/remove`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/http_routing_servers_remove

Код HTTP-ответа:

200 – в случае успеха;

404 – если сервер с таким идентификатором не существует

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <!--in-->
  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="id" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name = "out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:all>
              <xs:element name="ok"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <!--bad. command failed!-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/http_routing_servers_remove

```

<in>
  <request domain="biysk.local" id="server4"/>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="http_routing_servers_remove.xsd">
  <response>
    <ok/>
  </response>
</out>
```

[Hc_http_routing_servers_set_all](#)

Команда заменяет весь набор настроенных серверов HTTP маршрутизации новым набором.

Аналог команды в CoCon:

нет

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/http_routing_servers_set_all

Код HTTP-ответа:

200 – в случае успеха;

Xsd схема XML запроса/ответа

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="HttpServer">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="url" type="xs:string" use="required"/>
    <xs:attribute name="api_key" type="xs:string" use="optional"/>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="server" type="HttpServer" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:all>
              <xs:element name="ok"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <!--bad. command failed-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/http_routing_servers_set_all

```
<in>
  <request domain="biysk.local">
    <server id="serv4" url="http://localhost:8789" api_key="faih3ieheiqu9ath"/>
    <server id="serv3" url="192.168.2.21" api_key="oi6ohcheiw6et2ai"/>
    <server id="serv2" url="10.16.33.6" api_key="eeyaang4zo4jaemi"/>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
  <out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="http_routing_servers_set_all.xsd">
    <response>
      <ok/>
    </response>
  </out>
```

Hc_iface - Управление интерфейсами

- [Hc_iface_group_owner_info](#)
- [Hc_iface_group_owner_list](#)
- [Hc_iface_list](#)
- [Hc_iface_info](#)
- [Hc_iface_user_set](#)
- [Hc_iface_safe_user_set](#)

Hc_iface_group_owner_info

Команда позволяет вывести список свойств групп интерфейсов.

Могут быть следующие группы интерфейсов: сгруппированные по владельцу, по имени группы, по владельцу и имени группы, глобальная группа.

Аналог команды в CoCon:

```
/cluster/storage/<CLUSTER>/iface/group-info
/domain/<DOMAIN>/iface/group-info
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/iface_group_owner_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- propertyTypeType -->
  <xs:simpleType name="propertyTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="global"/>
      <xs:enumeration value="owner"/>
      <xs:enumeration value="group"/>
      <xs:enumeration value="group_owner"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- propertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute name="type" type="propertyTypeType" use="required"/>
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="owner" type="xs:string" use="optional"/>
            <xs:attribute name="group" type="xs:string" use="optional"/>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="header">
          <xs:complexType>
            <xs:attribute name="owner" type="xs:string" use="required"/>
            <xs:attribute name="group" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="properties">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```



```

        </xs:element>
    </xs:all>
</xs:complexType>
</xs:element>
</xs:schema>

```

Примеры:

Пример запроса информации о глобальной группе:

Запрос: http://192.168.1.21:9999/commands/iface_group_owner_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_group_
_owner_info.xsd">
    <request domain="biysk.local"/>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_group_o
wner_info.xsd">
    <header group="*" owner="*" />
    <properties/>
</out>

```

Пример запроса информации о группе по владельцу:

Запрос: http://192.168.1.21:9999/commands/iface_group_owner_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_group_
_owner_info.xsd">
    <request owner="sip1" domain="biysk.local"/>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_group_ow
ner_info.xsd">
  <header group="*" owner="sip1"/>
  <properties/>
</out>
```

Пример запроса информации о группе по имени группы:

Запрос: http://192.168.1.21:9999/commands/iface_group_owner_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_group
_owner_info.xsd">
  <request group="loc.gr" domain="biysk.local"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_group_ow
ner_info.xsd">
  <header group="loc.gr" owner="*" />
  <properties>
    <property type="group" key="gate" value="{gate_amqp, <<"acp.adapter.init.ex">>, <<"acp.sip.
sip1.ipset1.init.rk">>}" />
    <property type="group" key="group" value="{gate_amqp, <<"acp.adapter.init.ex">>, <<"acp.sip
.sip1.ipset1.init.rk">>}" />
    <property type="group" key="owner" value="sip1" />
  </properties>
</out>
```

Пример запроса информации о группе по имени группы и владельцу:

Запрос: http://192.168.1.21:9999/commands/iface_group_owner_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_group
_owner_info.xsd">
  <request owner="sip1" group="loc.gr" domain="biysk.local"/>
</in>
```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_group_o
wner_info.xsd">
  <header group="loc.gr" owner="sip1"/>
  <properties>
    <property type="group" key="gate"
value="{gate_amqp,&lt;&lt;&quot;acp.adapter.init.ex&quot;>>,&lt;&lt;&quot;acp.ecss10.sip1.init.rk
&quot;>>}" />
    <property type="group" key="owner" value="&quot;sip1&quot;" />
    <property type="group" key="terminal_type" value="basic" />
  </properties>
</out>

```

hc_iface_group_owner_list

Команда позволяет вывести список владельцев, каждый из которых содержит список групп интерфейсов, определенных для данного владельца.

Аналог команды в CoCon:

```
/cluster/storage/<CLUSTER>/iface/list
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/iface_group_owner_list

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- ownerType-->
  <xs:simpleType name="ownerTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="sip" />
      <xs:enumeration value="megaco" />
      <xs:enumeration value="virtual" />
      <xs:enumeration value="cc" />
      <xs:enumeration value="other" />
    </xs:restriction>
  </xs:simpleType>
  <!-- groupType-->
  <xs:complexType name="groupType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- ownerType-->
  <xs:complexType name="ownerType">
    <xs:sequence>
      <xs:element name="groups">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="group" type="groupType" minOccurs="0" maxOccurs="unbound"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="ownerTypeType" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="owners">

```

```

        <xs:complexType>
          <xs:sequence>
            <xs:element name="owner" type="ownerType" minOccurs="0" maxOccurs="un
bounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

В данном примере запрашиваются все пары {Owner, Group}, относящиеся к домену biysk.local.

Запрос:

http://192.168.1.21:9999/commands/iface_group_owner_list

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_iface
_group_owner_list.xsd">
  <request domain="biysk.local"></request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_iface_
group_owner_list.xsd">
  <owners>
    <owner name="sip1" type="sip">
      <groups>
        <group name="sip.ab"/>
      </groups>
    </owner>
    <owner name="megaco1" type="megaco">
      <groups/>
    </owner>
    <owner name=".virtual" type="other">
      <groups/>
    </owner>
    <owner name=".system_bridge" type="other">
      <groups/>
    </owner>
    <owner name=".system" type="other">
      <groups>
        <group name=".system"/>
      </groups>
    </owner>
  </owners>
</out>


```

hc_iface_list

Команда позволяет вывести список интерфейсов, определенных для конкретного домена.

Аналог команды в CoCon:

/domain/<DOMAIN>/iface/list

 Запросы на получение списка интерфейсов с использованием домена более ресурсоемкие, необходимо стараться избегать подобных запросов или выполнять во время простоя системы.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/iface_list

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;

- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- ifaceType -->
  <xs:complexType name="ifaceType">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="owner" type="xs:string" use="required"/>
    <xs:attribute name="group" type="xs:string" use="required"/>
    <xs:attribute name="type" type="xs:string" use="required"/>
    <xs:attribute name="active" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- ifacesType -->
  <xs:complexType name="ifacesType">
    <xs:sequence>
      <xs:element name="iface" type="ifaceType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="owner" type="xs:string" use="optional"/>
            <xs:attribute name="group" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="ifaces" type="ifacesType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Примеры:

Пример запроса всех интерфейсов в определенном домене:

Запрос: http://192.168.1.21:9999/commands/iface_list

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_iface
_list.xsd">
  <request domain="biysk.local"/>
</in>
```

Ответ:


```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_iface_list.xsd">
  <ifaces>
    <iface id="06201d997b9cd8f1" name="240616@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d99bedc3c0c" name="240466@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="true"/>
    <iface id="06201d997b9db807" name="240628@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d996cc0316f" name="240536@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="smg-4" name="smg-4" owner="sip1" group="smg.gr" type="sip" active="true"/>
    <iface id="06201d99d02a910b" name="240473@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="true"/>
    <iface id="06201d997bafec03" name="240772@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d997ba46191" name="240677@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d997baa029e" name="240754@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d996cbf2127" name="240508@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d997bb0095b" name="240777@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d996cc12fa0" name="240562@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d996cc03ffff" name="240538@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="to_aster" name="to_aster" owner="sip1" group="sbc.gr" type="sip" active="true"
/>
    <iface id="06201d997ba407d9" name="240663@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d996cc21da4" name="240587@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d997bb663d2" name="240792@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d997b9b03b0" name="240605@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d997ba42485" name="240667@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d995549d606" name="240002@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d997baf9e6" name="240769@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d997ba4a0d2" name="240684@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
  </ifaces>
</out>

```

```

    <iface id="06201d996cbefbb1" name="240502@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d996ccc00ea0" name="240530@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d996cc09448" name="240542@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="ems1" name="ems1" owner="sip1" group="ssw.gr" type="sip" active="true"/>
    <iface id="06201d996cbf4a88" name="240511@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="system:teleconference" name="system:teleconference" owner=".system"
group=".system" type="system" active="true"/>
    <iface id="06201d997ba16772" name="240647@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d996cbf0f2f" name="240505@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
    <iface id="06201d996cc223c5" name="240588@biysk.local" owner="sip1" group="loc.gr"
type="sip" active="false"/>
  </ifaces>
</out>

```

Пример запроса всех интерфейсов по определенной группе и домену:

Запрос: http://192.168.1.21:9999/commands/iface_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_iface_
_list.xsd">
  <request domain="biysk.local" group="sbc.gr"/>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_iface_l
ist.xsd">
  <ifaces>
    <iface id="nsk_sbc" name="nsk_sbc" owner="sip1" group="sbc.gr" type="sip" active="true"/>
    <iface id="to_aster" name="to_aster" owner="sip1" group="sbc.gr" type="sip" active="true"
/>
    <iface id="to_sipp" name="to_sipp" owner="sip1" group="sbc.gr" type="sip" active="true"/>
  </ifaces>
</out>

```

Пример запроса всех интерфейсов определенного владельца в определенной группе и домене:

Запрос: http://192.168.1.21:9999/commands/iface_list

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_iface_list.xsd">
  <request domain="biysk.local" owner="sip1" group="ssw.gr"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_iface_list.xsd">
  <ifaces>
    <iface id="bsk1" name="bsk1" owner="sip1" group="ssw.gr" type="sip" active="true"/>
    <iface id="bsk2" name="bsk2" owner="sip1" group="ssw.gr" type="sip" active="true"/>
    <iface id="ems1" name="ems1" owner="sip1" group="ssw.gr" type="sip" active="true"/>
    <iface id="ems2" name="ems2" owner="sip1" group="ssw.gr" type="sip" active="true"/>
  </ifaces>
</out>
```

hc_iface_info

Команда позволяет вывести список свойств определенного интерфейса, или noiface, если интерфейс не существует.

Аналог команды в CoCon:

```
/cluster/storage/<CLUSTER>/iface/info
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/iface_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- propertyTypeType-->
  <xs:simpleType name="propertyTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="global"/>
      <xs:enumeration value="owner"/>
      <xs:enumeration value="group"/>
      <xs:enumeration value="group_owner"/>
      <xs:enumeration value="iface"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="type" type="propertyTypeType" use="required"/>
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- ifaceRequestType-->
  <xs:complexType name="ifaceRequestType">
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="iface" type="ifaceRequestType" minOccurs="1" maxOccurs="unbounded"/
>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="property" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- ifaceResponseType-->
  <xs:complexType name="ifaceResponseType">
    <xs:all>
      <xs:element name="properties">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:all>
    <xs:attribute name="iface" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="owner" type="xs:string" use="required"/>
    <xs:attribute name="group" type="xs:string" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>

```

```

</xs:complexType>
<!-- noIfaceType-->
<xs:complexType name="noIfaceType">
  <xs:attribute name="iface" type="xs:string" use="required"/>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request" type="requestType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- out-->
<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="iface" type="ifaceResponseType" minOccurs="0" maxOccurs="unbound"/>
      <xs:element name="noIface" type="noIfaceType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример запроса информации о существующем интерфейсе:

Запрос: http://192.168.1.21:9999/commands/iface_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_iface_info.xsd">
  <request domain="biysk.local">
    <iface id="56089554a52e2f44" />
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_iface_info.xsd">
  <iface iface="smg-4" name="smg-4" owner="sip1" group="smg.gr" active="true">
    <properties>
      <property type="iface" key="media-profile" value="{media_profile,"default",user,
[media_profile_codecs,'<other>', [{media_profile_codec,<<"<other>">>,<<"*">>,true,#{}]],
#{offroad => false,'rtcp-enabled' => true}}, {media_profile_codecs,audio,
[media_profile_codec,<<"G722">>,<<"*">>,true,#{}}, {media_profile_codec,<<"PCMA">>,<<"*">>,true,
#{}, {media_profile_codec,<<"PCMU">>,<<"*">>,true,#{}}, {media_profile_codec,<<"G729">>,<<"*">>,
true,#{}}, {media_profile_codec,<<"telephone-event">>,<<"*">>,true,#{}}, #{offroad =>
false,'rtcp-enabled' => true}]],"#{}"/>
      <property type="iface" key="options_control" value="32000"/>
      <property type="iface" key="sipdomain" value="use_address"/>
      <property type="iface" key="stat\rejected\out" value="0"/>
      <property type="iface" key="stat\rejected\in" value="0"/>
      <property type="iface" key="stat\cps_max\total" value="5"/>
      <property type="iface" key="stat\cps_max\out" value="2"/>
      <property type="iface" key="stat\cps_max\in" value="5"/>
      <property type="iface" key="stat\out" value="160"/>
      <property type="iface" key="stat\in" value="5734"/>
      <property type="iface" key="ignore" value="none"/>
      <property type="iface" key="bandwidth\total" value="256"/>
      <property type="iface" key="adapter\version" value="3.14.0.702"/>
      <property type="iface" key="user_name" value="smg-4"/>
      <property type="iface" key="trunk" value="biysk.local.smg.gr.trunk.autoname"/>
      <property type="iface" key="subtype" value="trunk"/>
      <property type="iface" key="routing.context" value="ctx_from_ewsd"/>
      <property type="iface" key="req100rel" value="true"/>
      <property type="iface" key="remote_network_type" value="local"/>
      <property type="iface" key="remote_client" value="default"/>
      <property type="iface" key="port" value="5061"/>
      <property type="iface" key="network_interface" value="ipset1"/>
      <property type="iface" key="my_from" value="test.ssw.domain"/>
      <property type="iface" key="mode" value="sipt"/>
      <property type="iface" key="listen_port" value="5061"/>
      <property type="iface" key="ip_set" value="ipset1"/>
      <property type="iface" key="ip" value="192.168.2.8"/>
      <property type="iface" key="dynamic" value="false"/>
      <property type="iface" key="declaration" value="{admin,{1568,121155,720123}}"/>
      <property type="iface" key="cdpn_transformation" value="none"/>
      <property type="iface" key="cause_location" value="default"/>
      <property type="iface" key="profile" value="undefined"/>
      <property type="iface" key="type" value="sip"/>
      <property type="iface" key="id" value="smg-4"/>
      <property type="group" key="gate" value="{gate_amqp,<<"acp.adapter.init.ex">>,<<"acp.
sip.sip1.ipset1.init.rk">>}">
      <property type="group" key="domain" value="biysk.local"/>

```

```

        <property type="global" key="zone" value="<<"default">>" />
        <property type="global" key="terminal_type" value="smart" />
        <property type="global" key="site" value="<<"local">>" />
        <property type="global" key="dtmf_source" value="'rfc+info'" />
    </properties>
</iface>
</out>

```

Пример запроса информации о несуществующем интерфейсе:

Запрос: http://192.168.1.21:9999/commands/iface_info

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_iface_info.xsd">
  <request domain="biysk.local">
    <iface id="bla-bla" />
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_iface_info.xsd">
  <noIface iface="bla-bla" />
</out>

```

Ис_iface_user_set

Команда по установке свойств интерфейса.

Аналог команды в CoCon:

domain/<DOMAIN>/iface/user-set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/iface_user_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа `iface_user_set_common`:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="iface_user_set">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set ECSS iface(s) resources properties.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <!-- terminalTypeType-->
  <xs:simpleType name="terminalTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="basic"/>
      <xs:enumeration value="smart"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- dtmfSourceType-->
  <xs:simpleType name="dtmfSourceType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="info"/>
      <xs:enumeration value="rfc"/>
      <xs:enumeration value="rfc+info"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- terminalTypeType-->
  <xs:simpleType name="supportEncodingType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="utf8"/>
      <xs:enumeration value="latin1"/>
      <xs:enumeration value="cp1251"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- cfcSupportType-->
  <xs:simpleType name="cfcSupportType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="true"/>
      <xs:enumeration value="false"/>
    </xs:restriction>
  </xs:simpleType>

</xs:schema>

```

iface_user_set.xsd


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="iface_user_set">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set ECSS iface(s) resources properties.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="iface_user_set_common.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:token" use="required"/>
            <xs:attribute name="owner" type="xs:token" use="optional"/>
            <xs:attribute name="group" type="xs:token" use="optional"/>
            <xs:attribute name="ifaces" type="xs:normalizedString" use="required"/>
            <xs:attribute name="routing_context" type="xs:string" use="optional"/>
            <xs:attribute name="modifier" type="xs:string" use="optional"/>
            <xs:attribute name="site" type="xs:string" use="optional"/>
            <xs:attribute name="zone" type="xs:string" use="optional"/>
            <xs:attribute name="terminal_type" type="terminalTypeType" use="optional"
/>

            <xs:attribute name="dtmf_source" type="dtmfSourceType" use="optional"/>
            <xs:attribute name="support_encoding" type="supportEncodingType"
use="optional"/>

            <xs:attribute name="cfc_support" type="cfcSupportType" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:attribute name="status" type="xs:token"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

        </xs:choice>
    </xs:complexType>
</xs:element>
<!-- out -->

</xs:schema>

```

Пример запроса конкретного свойства:

Запрос:

http://192.168.1.21:9999/commands/iface_user_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_iface_info.xsd">
  <request domain="biysk.local" property="cfc_support">
    <iface id="ems1"/>
  </request>
</in>

```

Ответ: 200

```

status: 200
content:
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_iface_info.xsd">
  <iface iface="ems1" name="ems1" owner="" group="" active="false">
    <properties>
      <property type="global" key="cfc-support" value="true"/>
    </properties>
  </iface>
</out>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/iface_user_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="iface_user_set.xsd">
  <request domain="biysk.local" ifaces="064bc964f1c1f424" routing_context="default_context"/>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="iface_user
_set.xsd">
  <response status="ok"/>
</out>
```

Ис_iface_safe_user_set

Команда по установке основных свойств интерфейса.

Аналог команды в CoCon:

domain/<DOMAIN>/iface/user-set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/iface_safe_user_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" id="iface_user_set">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set ECSS iface(s) resources properties.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="iface_user_set_common.xsd" />
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:token" use="required"/>
            <xs:attribute name="owner" type="xs:token" use="optional"/>
            <xs:attribute name="group" type="xs:token" use="optional"/>
            <xs:attribute name="ifaces" type="xs:normalizedString" use="required"/>
            <xs:attribute name="call_record" type="xs:boolean" use="optional"/>
            <xs:attribute name="support_encoding" type="supportEncodingType"
use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:attribute name="status" type="xs:token"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- out -->
</xs:schema>

```

Пример:

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="iface_safe_user_set.xsd">
  <request domain="biysk.local" ifaces="06201d997ba16772" call_record="true"/>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="iface_safe_user_set.xsd">
  <response status="ok"/>
</out>
```

Hc_isup_cause_messages - управление текстовыми описаниями для ISUP cause

- [Hc_isup_cause_messages_clean](#)
- [Hc_isup_cause_messages_info](#)
- [Hc_isup_cause_messages_set](#)

[Hc_isup_cause_messages_clean](#)

Команда возвращает дефолтные значения текстовых описаний для ISUP cause-ов

Аналог команды в CoCon:

domain/<DOMAIN>/sip/isup-cause-messages/clean

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/isup_cause_messages_clean

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/isup_cause_messages_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_isup_cause_messages_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="code_1"/>
        <option name="external"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_isup_cause_messages_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="code_1"/>
        <success option="external"/>
      </domain>
    </domain_options>
  </result>
</out>

```

[Hc_isup_cause_messages_info](#)

Команда возвращает текстовые описания для ISUP cause-ов

Аналог команды в CoCon:

domain/<DOMAIN>/sip/isup-cause-messages/info

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/isup_cause_messages_info

Код http ответа

200 – в случае успеха;
404 – в случае ошибки.

Xsd схема XML запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="isupCauseMessagesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="code_1" type="xs:string" use="optional" />
        <xs:attribute name="code_2" type="xs:string" use="optional" />
        <xs:attribute name="code_3" type="xs:string" use="optional" />
        <xs:attribute name="code_4" type="xs:string" use="optional" />
        <xs:attribute name="code_5" type="xs:string" use="optional" />
        <xs:attribute name="code_6" type="xs:string" use="optional" />
        <xs:attribute name="code_7" type="xs:string" use="optional" />
        <xs:attribute name="code_8" type="xs:string" use="optional" />
        <xs:attribute name="code_9" type="xs:string" use="optional" />
        <xs:attribute name="code_16" type="xs:string" use="optional" />
        <xs:attribute name="code_17" type="xs:string" use="optional" />
        <xs:attribute name="code_18" type="xs:string" use="optional" />
        <xs:attribute name="code_19" type="xs:string" use="optional" />
        <xs:attribute name="code_20" type="xs:string" use="optional" />
        <xs:attribute name="code_21" type="xs:string" use="optional" />
        <xs:attribute name="code_22" type="xs:string" use="optional" />
        <xs:attribute name="code_23" type="xs:string" use="optional" />
        <xs:attribute name="code_25" type="xs:string" use="optional" />
        <xs:attribute name="code_26" type="xs:string" use="optional" />
        <xs:attribute name="code_27" type="xs:string" use="optional" />
        <xs:attribute name="code_28" type="xs:string" use="optional" />
        <xs:attribute name="code_29" type="xs:string" use="optional" />
        <xs:attribute name="code_30" type="xs:string" use="optional" />
        <xs:attribute name="code_31" type="xs:string" use="optional" />
        <xs:attribute name="code_34" type="xs:string" use="optional" />
        <xs:attribute name="code_38" type="xs:string" use="optional" />
        <xs:attribute name="code_39" type="xs:string" use="optional" />
        <xs:attribute name="code_40" type="xs:string" use="optional" />
        <xs:attribute name="code_41" type="xs:string" use="optional" />
        <xs:attribute name="code_42" type="xs:string" use="optional" />
        <xs:attribute name="code_43" type="xs:string" use="optional" />
        <xs:attribute name="code_44" type="xs:string" use="optional" />
        <xs:attribute name="code_46" type="xs:string" use="optional" />
        <xs:attribute name="code_47" type="xs:string" use="optional" />
        <xs:attribute name="code_49" type="xs:string" use="optional" />
        <xs:attribute name="code_50" type="xs:string" use="optional" />
        <xs:attribute name="code_53" type="xs:string" use="optional" />
        <xs:attribute name="code_55" type="xs:string" use="optional" />
        <xs:attribute name="code_57" type="xs:string" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

<xs:attribute name="code_58" type="xs:string" use="optional" />
<xs:attribute name="code_62" type="xs:string" use="optional" />
<xs:attribute name="code_63" type="xs:string" use="optional" />
<xs:attribute name="code_65" type="xs:string" use="optional" />
<xs:attribute name="code_66" type="xs:string" use="optional" />
<xs:attribute name="code_69" type="xs:string" use="optional" />
<xs:attribute name="code_70" type="xs:string" use="optional" />
<xs:attribute name="code_79" type="xs:string" use="optional" />
<xs:attribute name="code_81" type="xs:string" use="optional" />
<xs:attribute name="code_82" type="xs:string" use="optional" />
<xs:attribute name="code_83" type="xs:string" use="optional" />
<xs:attribute name="code_84" type="xs:string" use="optional" />
<xs:attribute name="code_85" type="xs:string" use="optional" />
<xs:attribute name="code_86" type="xs:string" use="optional" />
<xs:attribute name="code_87" type="xs:string" use="optional" />
<xs:attribute name="code_88" type="xs:string" use="optional" />
<xs:attribute name="code_90" type="xs:string" use="optional" />
<xs:attribute name="code_91" type="xs:string" use="optional" />
<xs:attribute name="code_95" type="xs:string" use="optional" />
<xs:attribute name="code_96" type="xs:string" use="optional" />
<xs:attribute name="code_97" type="xs:string" use="optional" />
<xs:attribute name="code_98" type="xs:string" use="optional" />
<xs:attribute name="code_99" type="xs:string" use="optional" />
<xs:attribute name="code_100" type="xs:string" use="optional" />
<xs:attribute name="code_101" type="xs:string" use="optional" />
<xs:attribute name="code_102" type="xs:string" use="optional" />
<xs:attribute name="code_103" type="xs:string" use="optional" />
<xs:attribute name="code_110" type="xs:string" use="optional" />
<xs:attribute name="code_111" type="xs:string" use="optional" />
<xs:attribute name="code_127" type="xs:string" use="optional" />
<xs:attribute name="external" type="xs:string" use="optional" />
</xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1"/>

```

```

        </xs:all>
        </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/isup_cause_messages_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_isup_cause_messages_info.xsd">
    <request>
        <domain_options domain="biysk.local" />
    </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_isup_cause_messages_info.xsd">
    <result>
        <domain_options>
            <system>
                <options xs:type="isupCauseMessagesOptionsType"/>
            </system>
            <domain name="biysk.local">
                <options xs:type="isupCauseMessagesOptionsType" external=""
code_99="language:default,text:Information element/parameter non-existent or not
implemented;language:en,text:Information element/parameter non-existent or not
implemented;language:ru,text:Несуществующий или нереализованный информационный элемент или
параметр" code_98="language:default,text:Message not compatible with call state or message type
non-existent;language:en,text:Message not compatible with call state or message type non-
existent;language:ru,text:Сообщение не соответствует состоянию вызова или сообщение
несуществующего типа или не реализовано" code_97="language:default,text:Message type non-existent

```

or not implemented;language:en,text:Message type non-existent or not implemented;language:ru,text:Несуществующий тип сообщения или не реализовано"

code_96="language:default,text:Mandatory information element is missing;language:en,text:Mandatory information element is missing;language:ru,text:Отсутствует обязательный информационный элемент" code_95="language:default,text:Invalid message unspecified;language:en,text:Invalid message unspecified;language:ru,text:Неправильное сообщение, без уточнения" code_91="language:default,text:Invalid transit network selection (national use);language:en,text:Invalid transit network selection (national use);language:ru,text:Неверный выбор транзитной сети" code_90="language:default,text:Non-existent CUG;language:en,text:Non-existent CUG;language:ru,text:Несуществующая замкнутая группа пользователей"

code_9="language:default,text:Preemption - circuit reserved for reuse;language:en,text:Preemption - circuit reserved for reuse;language:ru,text:Прерывание обслуживания вызова - канал зарезервирован для повторного использования" code_88="language:default,text:Incompatible destination;language:en,text:Incompatible destination;language:ru,text:Несовместимый пункт назначения" code_87="language:default,text:User not member of CUG;language:en,text:User not member of CUG;language:ru,text:Абонент не является членом замкнутой группы пользователей"

code_86="language:default,text:Call having the requested call identity has been cleared;language:en,text:Call having the requested call identity has been cleared;language:ru,text:Разъединение соединения для вызова, имеющего запрашиваемый идентификатор вызова" code_85="language:default,text:No call suspended;language:en,text:No call suspended;language:ru,text:Отсутствуют приостановленные вызовы"

code_84="language:default,text:Call identity in use;language:en,text:Call identity in use;language:ru,text:Использование индентификатора вызова" code_83="language:default,text:A suspended call exists but this call identity does not;language:en,text:A suspended call exists but this call identity does not;language:ru,text:Существующий вызов приостановлен, вызов не идентифицирован" code_82="language:default,text:Identified channel does not exist;language:en,text:Identified channel does not exist;language:ru,text:Выбранный канал не существует" code_81="language:default,text:Invalid call reference value;language:en,text:Invalid call reference value;language:ru,text:Неверное значение ссылки вызова"

code_8="language:default,text:Preemption;language:en,text:Preemption;language:ru,text:Прерывание обслуживания из-за вызова с более высоким приоритетом" code_79="language:default,text:Service or option not implemented unspecified;language:en,text:Service or option not implemented unspecified;language:ru,text:Услуга или опция не реализована, без уточнения"

code_70="language:default,text:Only restricted digital information bearer capability is available;language:en,text:Only restricted digital information bearer capability is available;language:ru,text:Доступна только ограниченная функция передачи цифровой информации"

code_7="language:default,text:Call awarded and being delivered in an established channel;language:en,text:Call awarded and being delivered in an established channel;language:ru,text:Вызов принят и передается по установленному каналу"

code_69="language:default,text:Requested facility not implemented;language:en,text:Requested facility not implemented;language:ru,text:Запрашиваемая услуга не реализована"

code_66="language:default,text:Channel type not implemented;language:en,text:Channel type not implemented;language:ru,text:Тип канала не реализован" code_65="language:default,text:Bearer capability not implemented;language:en,text:Bearer capability not implemented;language:ru,text:Функция передачи информации не реализована"

code_63="language:default,text:Service or option not available, unspecified;language:en,text:Service or option not available, unspecified;language:ru,text:Услуга или опция недоступна, без уточнения" code_62="language:default,text:Inconsistency in designated outgoing access information;language:en,text:Inconsistency in designated outgoing access information;language:ru,text:Несоответствие в информации назначенного исходящего доступа и абонентского класса" code_6="language:default,text:Channel unacceptable;language:en,text:Channel

unacceptable;language:ru,text:Канал не поддерживается" code_58="language:default,text:Bearer capability not presently available;language:en,text:Bearer capability not presently available;language:ru,text:Функция передачи информации в настоящее время недоступна" code_57="language:default,text:Bearer capability not authorized;language:en,text:Bearer capability not authorized;language:ru,text:Отсутствие авторизации для функции передачи информации" code_55="language:default,text:Incoming calls barred within CUG;language:en,text:Incoming calls barred within CUG;language:ru,text:Входящие вызовы ограничены в пределах замкнутой абонентской группы" code_53="language:default,text:Outgoing calls barred within CUG;language:en,text:Outgoing calls barred within CUG;language:ru,text:Исходящие вызовы ограничены в пределах замкнутой абонентской группы" code_50="language:default,text:Requested facility not subscribed;language:en,text:Requested facility not subscribed;language:ru,text:Отсутствует подписка на запрашиваемую услугу" code_5="language:default,text:Misdialled trunk prefix (national use);language:en,text:Misdialled trunk prefix (national use);language:ru,text:Неправильно набран префикс на транке" code_49="language:default,text:Quality of Service not available;language:en,text:Quality of Service not available;language:ru,text:Качество обслуживания недоступно" code_47="language:default,text:Resource unavailable unspecified;language:en,text:Resource unavailable unspecified;language:ru,text:Ресурс недоступен, без уточнения" code_46="language:default,text:Precedence call blocked;language:en,text:Precedence call blocked;language:ru,text:Заблокировано приоритетным вызовом" code_44="language:default,text:Requested circuit/channel not available;language:en,text:Requested circuit/channel not available;language:ru,text:Запрашиваемая канал не доступен" code_43="language:default,text:Access information discarded;language:en,text:Access information discarded;language:ru,text:Информация о доступе отклонена" code_42="language:default,text:Switching equipment congestion;language:en,text:Switching equipment congestion;language:ru,text:Перегрузка коммутационного оборудования" code_41="language:default,text:Temporary failure;language:en,text:Temporary failure;language:ru,text:Временная неисправность" code_40="language:default,text:Permanent frame mode connection operational;language:en,text:Permanent frame mode connection operational;language:ru,text:Действующее постоянное соединение в кадровом режиме" code_4="language:default,text:Send special information tone;language:en,text:Send special information tone;language:ru,text:Посылка специального звукового тона" code_39="language:default,text:Permanent frame mode connection out of service;language:en,text:Permanent frame mode connection out of service;language:ru,text:Постоянное соединение в кадровом режиме не обслуживается" code_38="language:default,text:Network out of order;language:en,text:Network out of order;language:ru,text:Неисправность сети" code_34="language:default,text:No circuit/channel available;language:en,text:No circuit/channel available;language:ru,text:Нет свободного канала" code_31="language:default,text:Normal unspecified;language:en,text:Normal unspecified;language:ru,text:Нормальное состояние, не уточнено" code_30="language:default,text:Response to STATUS ENQUIRY;language:en,text:Response to STATUS ENQUIRY;language:ru,text:Реакция на ЗАПРОС СОСТОЯНИЯ" code_3="language:default,text:No route to destination;language:en,text:No route to destination;language:ru,text:Нет маршрута до указанного номера" code_29="language:default,text:Facility rejected;language:en,text:Facility rejected;language:ru,text:Услуга отклонена" code_28="language:default,text:Invalid number format (address incomplete);language:en,text:Invalid number format (address incomplete);language:ru,text:Неверный формат номера (неполный адрес)" code_27="language:default,text:Destination out of order;language:en,text:Destination out of order;language:ru,text:Неисправность на конечном пункте" code_26="language:default,text:Non-selected user clearing;language:en,text:Non-selected user clearing;language:ru,text:Сброс невыбранного пользователя" code_25="language:default,text:Exchange - routing

```

error;language:en,text:Exchange - routing error;language:ru,text:Ошибка маршрутизации при обмене"
  code_23="language:default,text:Redirection to new destination;language:en,text:Redirection to
new destination;language:ru,text:Вызов перенаправлен" code_22="language:default,text:Number
changed;language:en,text:Number changed;language:ru,text:Номер изменен"
code_21="language:default,text:Call rejected;language:en,text:Call
rejected;language:ru,text:Вызов отклонен" code_20="language:default,text:Subscriber
absent;language:en,text:Subscriber absent;language:ru,text:Абонент недоступен"
code_2="language:default,text:No route to specified transit network (national
use);language:en,text:No route to specified transit network (national use);language:ru,text:Нет
маршрута к указанной транзитной сети" code_19="language:default,text:No answer from user (user
alerted);language:en,text:No answer from user (user alerted);language:ru,text:Нет ответа от
абонента (абонент уведомлен)" code_18="language:default,text:No user
responding;language:en,text:No user responding;language:ru,text:Абонент не отвечает"
code_17="language:default,text>User busy;language:en,text>User busy;language:ru,text:Абонент
занят" code_16="language:default,text:Normal call clearing;language:en,text:Normal call
clearing;language:ru,text:Нормальное завершение вызова"
code_127="language:default,text:Interworking unspecified;language:en,text:Interworking
unspecified;language:ru,text:Взаимодействие, без уточнения"
code_111="language:default,text:Protocol error unspecified;language:en,text:Protocol error
unspecified;language:ru,text:Ошибка протокола, без уточнения"
code_110="language:default,text:Message with unrecognized parameter
discarded;language:en,text:Message with unrecognized parameter
discarded;language:ru,text:Сообщение с нераспознанным параметром, отбрасывается"
code_103="language:default,text:Parameter non-existent or not implemented - passed on
(national;language:en,text:Parameter non-existent or not implemented - passed on
(national;language:ru,text:Параметр не существует или не реализован"
code_102="language:default,text:Recovery on timer expiry;language:en,text:Recovery on timer
expiry;language:ru,text:Восстановление по истечению временного таймера"
code_101="language:default,text:Message not compatible with call state;language:en,text:Message
not compatible with call state;language:ru,text:Сообщение не соответствует состоянию вызова"
  code_100="language:default,text:Invalid information element contents;language:en,text:Invalid
information element contents;language:ru,text:Неверное содержание информационного элемента"
  code_1="language:default,text:Unallocated (unassigned) number;language:en,text:Unallocated
(unassigned) number;language:ru,text:Номер не найден (не существует)"/>
    </domain>
  </domain_options>
</result>
</out>

```

Ис_isup_cause_messages_set

Команда задает значения текстовых описаний для ISUP cause-ов

Аналог команды в CoSop:

domain/<DOMAIN>/sip/isup-cause-messages/set

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/isup_cause_messages_set

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="isupCauseMessagesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="code_1" type="xs:string" use="optional" />
        <xs:attribute name="code_2" type="xs:string" use="optional" />
        <xs:attribute name="code_3" type="xs:string" use="optional" />
        <xs:attribute name="code_4" type="xs:string" use="optional" />
        <xs:attribute name="code_5" type="xs:string" use="optional" />
        <xs:attribute name="code_6" type="xs:string" use="optional" />
        <xs:attribute name="code_7" type="xs:string" use="optional" />
        <xs:attribute name="code_8" type="xs:string" use="optional" />
        <xs:attribute name="code_9" type="xs:string" use="optional" />
        <xs:attribute name="code_16" type="xs:string" use="optional" />
        <xs:attribute name="code_17" type="xs:string" use="optional" />
        <xs:attribute name="code_18" type="xs:string" use="optional" />
        <xs:attribute name="code_19" type="xs:string" use="optional" />
        <xs:attribute name="code_20" type="xs:string" use="optional" />
        <xs:attribute name="code_21" type="xs:string" use="optional" />
        <xs:attribute name="code_22" type="xs:string" use="optional" />
        <xs:attribute name="code_23" type="xs:string" use="optional" />
        <xs:attribute name="code_25" type="xs:string" use="optional" />
        <xs:attribute name="code_26" type="xs:string" use="optional" />
        <xs:attribute name="code_27" type="xs:string" use="optional" />
        <xs:attribute name="code_28" type="xs:string" use="optional" />
        <xs:attribute name="code_29" type="xs:string" use="optional" />
        <xs:attribute name="code_30" type="xs:string" use="optional" />
        <xs:attribute name="code_31" type="xs:string" use="optional" />
        <xs:attribute name="code_34" type="xs:string" use="optional" />
        <xs:attribute name="code_38" type="xs:string" use="optional" />
        <xs:attribute name="code_39" type="xs:string" use="optional" />
        <xs:attribute name="code_40" type="xs:string" use="optional" />
        <xs:attribute name="code_41" type="xs:string" use="optional" />
        <xs:attribute name="code_42" type="xs:string" use="optional" />
        <xs:attribute name="code_43" type="xs:string" use="optional" />
        <xs:attribute name="code_44" type="xs:string" use="optional" />
        <xs:attribute name="code_46" type="xs:string" use="optional" />
        <xs:attribute name="code_47" type="xs:string" use="optional" />
        <xs:attribute name="code_49" type="xs:string" use="optional" />
        <xs:attribute name="code_50" type="xs:string" use="optional" />
        <xs:attribute name="code_53" type="xs:string" use="optional" />
        <xs:attribute name="code_55" type="xs:string" use="optional" />
        <xs:attribute name="code_57" type="xs:string" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```



```

<xs:attribute name="code_58" type="xs:string" use="optional" />
<xs:attribute name="code_62" type="xs:string" use="optional" />
<xs:attribute name="code_63" type="xs:string" use="optional" />
<xs:attribute name="code_65" type="xs:string" use="optional" />
<xs:attribute name="code_66" type="xs:string" use="optional" />
<xs:attribute name="code_69" type="xs:string" use="optional" />
<xs:attribute name="code_70" type="xs:string" use="optional" />
<xs:attribute name="code_79" type="xs:string" use="optional" />
<xs:attribute name="code_81" type="xs:string" use="optional" />
<xs:attribute name="code_82" type="xs:string" use="optional" />
<xs:attribute name="code_83" type="xs:string" use="optional" />
<xs:attribute name="code_84" type="xs:string" use="optional" />
<xs:attribute name="code_85" type="xs:string" use="optional" />
<xs:attribute name="code_86" type="xs:string" use="optional" />
<xs:attribute name="code_87" type="xs:string" use="optional" />
<xs:attribute name="code_88" type="xs:string" use="optional" />
<xs:attribute name="code_90" type="xs:string" use="optional" />
<xs:attribute name="code_91" type="xs:string" use="optional" />
<xs:attribute name="code_95" type="xs:string" use="optional" />
<xs:attribute name="code_96" type="xs:string" use="optional" />
<xs:attribute name="code_97" type="xs:string" use="optional" />
<xs:attribute name="code_98" type="xs:string" use="optional" />
<xs:attribute name="code_99" type="xs:string" use="optional" />
<xs:attribute name="code_100" type="xs:string" use="optional" />
<xs:attribute name="code_101" type="xs:string" use="optional" />
<xs:attribute name="code_102" type="xs:string" use="optional" />
<xs:attribute name="code_103" type="xs:string" use="optional" />
<xs:attribute name="code_110" type="xs:string" use="optional" />
<xs:attribute name="code_111" type="xs:string" use="optional" />
<xs:attribute name="code_127" type="xs:string" use="optional" />
<xs:attribute name="external" type="xs:string" use="optional" />
</xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

        </xs:complexType>
    </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
            <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/isup_cause_messages_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_isup_cause_messages_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="isupCauseMessagesOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="isupCauseMessagesOptionsType" external="666:language:en,text:New
desc;language:ru,text:Новое описание|667:language:en,text:Desc"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_isup_cause_messages_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="external"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Hc_ivr_blocks_outputs_get - Команда просмотра блоков IVR-скрипта

Hc_ivr_blocks_outputs_get

Команда предназначена для просмотра блоков IVR-скрипта.

Аналог команды в CoSop:

domain/<DOMAIN>/ivr/script/get_outputs

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/ivr_blocks_outputs_get

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- scriptType -->
  <xs:complexType name="scriptType">
    <xs:sequence>
      <xs:element name="block" type="blockType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- scriptType -->
  <xs:complexType name="blockType">
    <xs:attribute name="name" type="xs:string"/>
    <xs:attribute name="output" type="xs:string"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="script_id" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:sequence maxOccurs="unbounded" minOccurs="0">
        <xs:element name="script" type="scriptType"/>
      </xs:sequence>
      <xs:element name="error" maxOccurs="1" type="errorType"/>
    </xs:choice>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/ivr_blocks_outputs_get

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_ivr_b
locks_outputs_get.xsd">
  <request domain="biysk.local" script_id="062029e4e7ca8ffe"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ivr_blo
cks_outputs_get.xsd">
  <response>
    <script id="062029e4e7ca8ffe" name="dial_to_cc">
      <block name="begin_1" output="true"/>
      <block name="queue-cc_2" output="Full"/>
      <block name="queue-cc_2" output="Error"/>
      <block name="queue-cc_2" output="Ok"/>
    </script>
  </response>
</out>
```

Hc_ivr_script_restriction - Управление ограничениями IVR скриптов

- [Hc_ivr_scripts_restriction_profile_info](#)
- [Hc_ivr_scripts_restriction_profile_apply](#)
- [Hc_ivr_scripts_restriction_profile_set](#)
- [Hc_ivr_scripts_restriction_profile_remove](#)
- [Hc_ivr_scripts_restriction_options_info](#)
- [Hc_ivr_scripts_restriction_options_set](#)

Hc_ivr_scripts_restriction_profile_info

Команда для просмотра профилей ограничений IVR скриптов по имени.

Замечание: команда доступна, если в лицензии открыта опция "IVR script's customization"**Аналог команды в CoCon:****/system/ivr/script/restrictions/info****Метод HTTP-запроса:**

POST

Шаблон HTTP-запроса:

http://host:port/commands/ivr_scripts_restriction_profile_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- profileType -->
  <xs:complexType name="profileType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
    <xs:attribute name="max_blocks_per_script" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Max IVR blocks per one IVR script. Available values:
          infinity - no restrictions on Max IVR blocks;
          [0, 16383] - max count if the IVR blocks.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="ivr_script_limit" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Max IVR scripts per domain. Available values:
          infinity - no restrictions on Max IVR scripts;
          [0, 65536] - max count if the IVR blocks.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="available_blocks" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The list of the available IVR blocks splitted by comma.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="asr" type="xs:boolean" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Automatic speech recognition.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="profile" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="profile" type="profileType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

```

```

</xs:complexType>
<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request" type="requestType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="response" type="responseType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1

Запрос:

http://192.168.1.21:9999/commands/ivr_scripts_restriction_profile_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ivr_scripts_restriction_profile_info.xsd">
  <request profile="ivr_level1"/>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ivr_scripts_restriction_profile_info.xsd">
  <response>
    <profile name="ivr_level1" description="" max_blocks_per_script="infinity"
      ivr_script_limit="12" available_blocks="*" asr="false"/>
  </response>
</out>

```

Пример 2

Запрос:


```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ivr_scripts_restriction_profile_info.xsd">
  <request/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ivr_scripts_restriction_profile_info.xsd">
  <response>
    <profile name="ivr_level1" description="" max_blocks_per_script="infinity"
ivr_script_limit="12" available_blocks="*" asr="false"/>
    <profile name="скрипт4" description="Описание 4" max_blocks_per_script="8"
ivr_script_limit="10" available_blocks="ivr,time,info" asr="false"/>
    <profile name="test4" description="test4" max_blocks_per_script="8" ivr_script_limit="10"
available_blocks="ivr,time,info,play" asr="false"/>
    <profile name="super_ivr" description="" max_blocks_per_script="infinity"
ivr_script_limit="100" available_blocks="*" asr="false"/>
  </response>
</out>
```

[Hc_ivr_scripts_restriction_profile_apply](#)

Команда применения профиля ограничений IVR скриптов на созданный домен.

Замечание: команда доступна, если в лицензии открыта опция "IVR script's customization".

Аналог команды в CoCon:

`/system/ivr/script/restrictions/apply`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ivr_scripts_restriction_profile_apply

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="error" type="errorType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/ivr_scripts_restriction_profile_apply

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ivr_scripts_restriction_profile_apply.xsd">
  <request name="super_ivr" domain="biysk.local"/>
</in>

```

Ответ:

201

Ис_ivr_scripts_restriction_profile_set

Команда для создания и/или изменения профилей ограничений IVR скриптов.

Замечание: команда доступна, если в лицензии открыта опция "IVR script's customization"**Аналог команды в CoCon:**

`/system/ivr/script/restrictions/set`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ivr_scripts_restriction_profile_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- profileType -->
  <xs:complexType name="profileType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="max_blocks_per_script" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Max IVR blocks per one IVR script. Available values:
          infinity - no restrictions on Max IVR blocks;
          [0, 16383] - max count if the IVR blocks.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="ivr_script_limit" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Max IVR scripts per domain. Available values:
          infinity - no restrictions on Max IVR scripts;
          [0, 65536] - max count if the IVR blocks.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="available_blocks" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The list of the available IVR blocks splitted by comma.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="asr" type="xs:boolean" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Automatic speech recognition.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>

```

```

<!-- requestType -->
<xs:complexType name="requestType">
  <xs:sequence>
    <xs:element name="profile" type="profileType" minOccurs="1" maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>
<!-- responseType -->
<xs:complexType name="responseType">
  <xs:sequence>
    <xs:element name="ok" type="okType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request" type="requestType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="response" type="responseType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример**Запрос:**

http://192.168.1.21:9999/commands/ivr_scripts_restriction_profile_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ivr_scripts_restriction_profile_set.xsd">
  <request>
    <profile name="ivr_level1" max_blocks_per_script="infinity" available_blocks=""
ivr_script_limit="12"/>
    <profile name="скрипт4" description="Описание 4" max_blocks_per_script="8"
available_blocks="ivr,time,info" ivr_script_limit="10"/>
  </request>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ivr_scripts_restriction_profile_set.xsd">
  <response>
    <ok name="ivr_level1"/>
    <ok name="скрипт4"/>
  </response>
</out>
```

[Hc_ivr_scripts_restriction_profile_remove](#)

Команда для удаления профилей ограничений IVR скриптов по имени.

Замечание: команда доступна, если в лицензии открыта опция "IVR script's customization".

Аналог команды в CoCon:

`/system/ivr/script/restrictions/remove`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ivr_scripts_restriction_profile_remove

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- profileType -->
  <xs:complexType name="profileType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="profile" type="profileType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="ok" type="okType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/ivr_scripts_restriction_profile_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ivr_scripts_restriction_profile_remove.xsd">
  <request>
    <profile name="ivr_level5"/>
    <profile name="скрипт4"/>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ivr_scripts_restriction_profile_remove.xsd">
  <response>
    <ok name="script4"/>
    <ok name="скрипт4"/>
    <error name="ivr_level5" reason="not_exists"/>
  </response>
</out>
```

[Hc_ivr_scripts_restriction_options_info](#)

Команда для просмотра ограничений IVR скриптов на уровне домена

Аналог команды в CoCon:

domain/<DOMAIN>/ivr/script/restrictions/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ivr_scripts_restriction_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show IVR scripts restriction options.
      Usage: http://server:9999/commands/ivr\_scripts\_restriction\_options\_info.
      Implementation: hc\_ivr\_scripts\_restriction\_options\_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2017, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="ivrScriptsRestrictionOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="max_blocks_per_script" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Max IVR blocks per one IVR script. Available values:
              infinity - no restrictions on Max IVR blocks;
              [0, 16383] - max count if the IVR blocks.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="ivr_script_limit" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Max IVR scripts per domain. Available values:
              infinity - no restrictions on Max IVR scripts;
              [0, 65536] - max count if the IVR blocks.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="available_blocks" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              The list of the available IVR blocks splitted by comma.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="asr" type="xs:boolean" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Automatic speech recognition.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

        </xs:annotation>
        </xs:attribute>
    </xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
    <xs:all>
        <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/ivr_scripts_restriction_options_info

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ivr_scripts_restriction_options_info.xsd">
    <request>
        <domain_options domain="biysk.local"/>
    </request>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ivr_scripts_restriction_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="ivrScriptsRestrictionOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="ivrScriptsRestrictionOptionsType"
max_blocks_per_script="infinity" ivr_script_limit="100" available_blocks="" asr="false"/>
      </domain>
    </domain_options>
  </result>
</out>
```

[Hc_ivr_scripts_restriction_options_set](#)

Команда для редактирования ограничений IVR скриптов на уровне домена

Аналог команды в CoСon:

domain/<DOMAIN>/ivr/script/restrictions/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ivr_scripts_restriction_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set IVR scripts restriction options.
      Usage: http://server:9999/commands/ivr_scripts_restriction_options_set.
      Implementation: hc_ivr_scripts_restriction_options_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2017, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="ivrScriptsRestrictionOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="max_blocks_per_script" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Max IVR blocks per one IVR script. Available values:
              infinity - no restrictions on Max IVR blocks;
              [0, 16383] - max count if the IVR blocks.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="ivr_script_limit" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Max IVR scripts per domain. Available values:
              infinity - no restrictions on Max IVR scripts;
              [0, 65536] - max count if the IVR blocks.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="available_blocks" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              The list of the available IVR blocks splitted by comma.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="asr" type="xs:boolean" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Automatic speech recognition.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

        </xs:attribute>
    </xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
    <xs:all>
        <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainSetInType"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
            <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/ivr_scripts_restriction_options_set

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ivr_scripts_restriction_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="ivrScriptsRestrictionOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="ivrScriptsRestrictionOptionsType" max_blocks_per_script="10"
ivr_script_limit="6" available_blocks="begin,play,connect,dial,queue"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ivr_scripts_restriction_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="available_blocks"/>
        <success option="ivr_script_limit"/>
        <success option="max_blocks_per_script"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Ис_ivr_script - Управление IVR скриптами

- [Ис_ivr_script_delete](#)
- [Ис_ivr_script_import](#)
- [Ис_ivr_script_list](#)
- [Ис_ivr_script_show](#)
- [Ис_ivr_script_validate](#)
- [Ис_ivr_script_rename](#)

Ис_ivr_script_delete

Команда позволяет удалить IVR-скрипт с определенным именем из определенного домена.

Аналог команды в CoCon:

`domain/<DOMAIN>/ivr/script/delete`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ivr_script_delete

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- okReasonType -->
  <xs:simpleType name="okReasonType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="success"/>
      <xs:enumeration value="reset_to_default"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="script_id" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="okReasonType" use="optional"/>
  </xs:complexType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="script_id" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- scriptType -->
  <xs:complexType name="scriptType">
    <xs:attribute name="script_id" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="script" type="scriptType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="ok" type="okType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>

```



```

        <xs:all>
            <xs:element name="response" type="responseType"/>
        </xs:all>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/ivr_script_delete

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_ivr_script_delete.xsd">
  <request domain="biysk.local">
    <script script_id="edf3acb838c34fde"/>
    <script script_id="edf3a7cd39752236"/>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ivr_script_delete.xsd">
  <response>
    <ok script_id="edf3acb838c34fde"/>
    <ok script_id="edf3a7cd39752236"/>
  </response>
</out>

```

[Hc_ivr_script_import](#)

Команда позволяет импортировать IVR-скрипт в систему.

Аналог команды в CoCon:

```
/domain/<DOMAIN>/ivr/script/.import
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

```
http://host:port/commands/ivr_script_import
```

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="script_id" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="script_id" type="xs:string" use="optional"/>
    <xs:attribute name="body" type="xs:base64Binary" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok" type="okType" maxOccurs="unbounded"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример 1

Создание нового скрипта

Запрос:

http://192.168.1.21:9999/commands/ivr_script_import


```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_ivr_script_import.xsd">
  <request domain="p.city" script_id="script1"
body=
"eyJhY3Rpb25zIjpw7I1N2Z2pzRzEwMTEiOmsibmFtZSI6InJpbmciLCJwYXJhbXMiOnt9LCJwb3MiOmsieCI6MSwieSI6MH0s
ImNhc2VzIjpw7Im5leHQiOiJTdmdqc0cxMDMzIn19LCJTdmdqc0cxMDMzIjpw7Im5hbWUiOiJjYXNlcyI6eyJuZWh0IjoU3ZnanNHMTI0N
yI6e30sInBvcyI6eyJ4IjoxLCJ5IjoxfSwiY2FzZXMiOmsibmV4dCI6I1N2Z2pzRzEwNjQifX0sI1N2Z2pzRzEwNjQifX0sibm
FtZSI6Im12ciIsInBhcmFtcyI6eyJpdnIiOlt7ImNvbW1hbmQiOiIqIiwZGVzY3JpcHRpb24iOiIifV19LCJwb3MiOmsieCI6MSwieSI6Mn0sImNhc2VzIjpw7IioiOiJTdmdqc0cxMTA5IiwRW1wdHkiOiJTdmdqc0cxMTM5In19LCJTdmdqc0cxMTA5Ijpw7
Im5hbWUiOiJwbGF5IiwicGFyYW1zIjpw7fSwicG9zIjpw7IngiOjAsInkiOjN9LCJjYXNlcyI6eyJuZWh0IjoU3ZnanNHMTI0N
CJ9fSwiU3ZnanNHMTI0SI6eyJuYw1lIjoizGlnaXRtYXAiLCJwYXJhbXMiOnt9LCJwb3MiOmsieCI6MSwieSI6M30sImNhc2
VzIjpw7I1llcyI6I1N2Z2pzRzExODQiLCJ0byI6I1N2Z2pzRzEwMTQifX0sI1N2Z2pzRzExODQiOmsibmFtZSI6InNhbWUtY2F
sbCIsInBhcmFtcyI6e30sInBvcyI6eyJ4IjoxLCJ5IjoxfSwiY2FzZXMiOnt9fSwiU3ZnanNHMTI0SI6eyJuYw1lIjoicmVj
IiwicGFyYW1zIjpw7fSwicG9zIjpw7IngiOjIsInkiOjR9LCJjYXNlcyI6eyJuZWh0IjoU3ZnanNHMTI0SI6eyJuYw1lIjoicmVj
TI0NCI6eyJuYw1lIjoicmVjY2UtbWpbcCIsInBhcmFtcyI6e30sInBvcyI6eyJ4IjowLCJ5IjoxfSwiY2FzZXMiOnt9fSwiU3
ZnanNHMTI0SI6eyJuYw1lIjoizZ90byIsInBhcmFtcyI6eyJnb3RvIjoU3ZnanNHMTA2NCIsIm1heF9ob3BzIjoxfSwicG9
zIjpw7IngiOjIsInkiOjV9LCJjYXNlcyI6eyJuZWh0IjoU3ZnanNHMTA2NCJ9fX19"/>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_ivr_script_import.xsd">
  <ok script_id="script1" />
</out>
```

Пример 3 (ошибочный)

Запрос:

http://192.168.23.34:9999/commands/ivr_script_import

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_ivr_script_import.xsd">
  <request domain="biysk.local" script_id="script1"
body=
"eyJhY3Rpb25zIjpw7I1N2Z2pzRzEwMTEiOmsibmFtZSI6InJpbmciLCJwYXJhbXMiOnt9LCJwb3MiOmsieCI6MSwieSI6MH"/
>
</in>
```

Код ответа:

404

Ответ:

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ivr_script_import.xsd">
  <error cmd="ivr_script_import" reason="invalid_input_xml" body="{invalid_xml,['Parse xml
error: {1,xmerl_b64Bin,['syntax error before: \",[\'$end\'\"]']}]}" entity="<?xml version='1.0'
encoding='latin1'?> <in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ivr_script_import.xsd"> <request domain="biysk.local"
script_id="script1" body="eyJhY3Rpb25zIjpb7I1N2Z2pzRzEwMTEiOnsibmFtZSI6InJpbmciLCJwYXJhbXMiOnt9LCJ
wb3MiOnsieCI6MSwieSI6MH"/> </in>"/>
  </out>
```

Ис_ivr_script_list

Команда просмотра списка IVR-скриптов в определенном домене.

Аналог команды в CoCon:

domain/<DOMAIN>/ivr/script/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ivr_script_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- Types -->
  <xs:simpleType name="directionType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="incoming"/>
      <xs:enumeration value="outgoing"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="ownerType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="system"/>
      <xs:enumeration value="domain"/>
      <xs:enumeration value="user"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- scriptType -->
  <xs:complexType name="scriptType">
    <xs:attribute name="script_id" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="direction" type="directionType" use="required"/>
    <xs:attribute name="owner" type="ownerType" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="script" type="scriptType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/ivr_script_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ivr_script_list.xsd">
  <request domain="biysk.local"/>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ivr_script_list.xsd">
  <response>
    <script script_id="062029e4e7ca8ffe" name="dial_to_cc" description=""
direction="incoming" owner="user"/>
    <script script_id="enter_number_of_teleconference" name="enter_number_of_teleconference"
description="Скрипт ввода номера телеконференции. Номер телеконференции вводится в блоке ivr.
Вход в телеконференции осуществляется блоком dial" direction="incoming" owner="domain"/>
    <script script_id="default_incoming_call" name="default_incoming_call"
description="Скрипт для обработки всех входящих вызовов в домен и маршрутизации их через IVR"
direction="incoming" owner="domain"/>
    <script script_id="06202b326f2fd9e8" name="dialer" description="" direction="incoming"
owner="user"/>
    <script script_id="06260fa4592a4419" name="to_cc" description="" direction="incoming"
owner="user"/>
    <script script_id="0622071a81711369" name="custom_cb_out" description=""
direction="outgoing" owner="user"/>
    <script script_id="fax_to_email" name="fax_to_email" description="Приём факса и отправка
на email, настроенный на уровне алиаса" direction="incoming" owner="domain"/>
    <script script_id="enter_number_of_conference_room"
name="enter_number_of_conference_room" description="Скрипт ввода номера комнаты конференции.
Номер комнаты конференции вводится в блоке ivr. Вход в комнату конференции осуществляется блоком
dial." direction="incoming" owner="domain"/>
  </response>
</out>

```

Ис_ivr_script_show

Команда просмотра конфигурации IVR-скрипта в определенном домене.

Аналог команды в CoCon:

`domain/<DOMAIN>/ivr/script/show`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ivr_script_show

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- ownerType -->
  <xs:simpleType name="ownerType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="system"/>
      <xs:enumeration value="domain"/>
      <xs:enumeration value="user"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- scriptType -->
  <xs:complexType name="scriptType">
    <xs:attribute name="script_id" type="xs:string" use="required"/>
    <xs:attribute name="owner" type="ownerType" use="required"/>
    <xs:attribute name="body" type="xs:base64Binary" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="script_id" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice minOccurs="1" maxOccurs="1">
      <xs:element name="script" type="scriptType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1


```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_ivr_script_show.xsd">
  <request domain="biysk.local" script_id="123456789"/>
</in>
```

Ответ: 404

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="common_error.xsd">
  <error cmd="ivr_script_show" reason="error" body="<?xml version="1.0"?><out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_ivr_script_show.xsd"><response><error cmd="ivr_script_show"
  reason="unknown_error" body="not_found"/></response></out>"/>
</out>
```

Ис_ivr_script_validate

Команда позволяет проверить валидность IVR-скрипта в системе.

Аналог команды в CoCon:

domain/<DOMAIN>/ivr/script/.validate

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ivr_script_validate

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- okType -->
  <xs:complexType name="okType"/>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="body" type="xs:base64Binary" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok" type="okType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1

Запрос:

http://192.168.1.21:9999/commands/ivr_script_validate

404

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_ivr_script_validate.xsd">
  <response>
    <error cmd="ivr_script_validate" reason="unknown_error" body="bad_script"/>
  </response>
</out>
```

Ис_ивр_скрипт_реименование

Команда позволяет переименовать IVR-скрипт в определенном домене.

Аналог команды в CoCon:

domain/<DOMAIN>/ivr/script/rename

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/ivr_script_rename

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="script_id" type="xs:string" use="required"/>
    <xs:attribute name="new_name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="script_id" type="xs:string" use="required"/>
    <xs:attribute name="new_name" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- scriptType -->
  <xs:complexType name="scriptType">
    <xs:attribute name="script_id" type="xs:string" use="required"/>
    <xs:attribute name="new_name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="script" type="scriptType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="ok" type="okType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```
</xs:schema>
```

Пример 1

Запрос:

http://192.168.1.21:9999/commands/ivr_script_rename

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_ivr_script_rename.xsd">
  <request domain="biysk.local">
    <script script_id="edf3a248c0fbcfde" new_name="new_script_1"/>
    <script script_id="edf3974226aaf4e9" new_name="to_cc2"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ivr_script_rename.xsd">
  <response>
    <ok script_id="edf3a248c0fbcfde" new_name="new_script_1"/>
    <ok script_id="edf3974226aaf4e9" new_name="to_cc2"/>
  </response>
</out>
```

hc_ivr_variables - Команды просмотра списка переменных IVR-скрипта

- [hc_ivr_undeclared_variables_get](#)
- [hc_ivr_declared_variables_get](#)

hc_ivr_undeclared_variables_get

Команда предназначена для просмотра списка переменных IVR-скрипта.

Аналог команды в CoCon:

domain/<DOMAIN>/ivr/script/get_variables

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/ivr_undeclared_variables_get

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- variableType -->
  <xs:complexType name="variableType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="script_id" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:sequence maxOccurs="unbounded">
        <xs:element name="variable" type="variableType"/>
      </xs:sequence>
      <xs:element name="error" maxOccurs="1" type="errorType"/>
    </xs:choice>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Примеры:

Запрос:

http://192.168.1.21:9999/commands/ivr_undeclared_variables_get

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_ivr_undeclared_variables_get.xsd">
  <request domain="biysk.local" script_id="334e5ceba166ae29"/>
</in>
```

Ответ, если переменных нет:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ivr_undeclared_variables_get.xsd">
  <response/>
</out>
```

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ivr_undeclared_variables_get.xsd">
  <response>
    <variable name="Variable1"/>
    <variable name="Variable2"/>
  </response>
</out>
```

hc_ivr_declared_variables_get

Команда предназначена для просмотра списка задекларированных переменных IVR-скрипта.

Аналог команды в CoCon:**domain/<DOMAIN>/ivr/script/get_variables****Метод HTTP-запроса:**

POST

Шаблон HTTP-запроса:http://server:9999/commands/ivr_declared_variables_get**Код HTTP-ответа:**

- 200 – в случае успеха;

- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- variableType -->
  <xs:complexType name="variableType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="script_id" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:sequence maxOccurs="unbounded">
        <xs:element name="variable" type="variableType"/>
      </xs:sequence>
      <xs:element name="error" maxOccurs="1" type="errorType"/>
    </xs:choice>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Примеры:

Запрос:

http://192.168.1.21:9999/commands/ivr_declared_variables_get

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_ivr_declared_variables_get.xsd">
  <request domain="biysk.local" script_id="06260ff97381766d"/>
</in>
```

Ответ, если переменных нет:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ivr_declared_variables_get.xsd">
  <response/>
</out>
```

Нс_licence - Управление лицензиями (HTTP-терминал)

- [Нс_add_licence](#)
- [Нс_current_limits](#)
- [Нс_list_licence](#)
- [Нс_delete_licence](#)
- [Нс_set_passport](#)
- [Нс_show_passport](#)

Нс_add_licence

Команда добавляет новую лицензию в базу данных ECSS-10.

Аналог команды в CoCon:

cluster/storage/<CLUSTER>/licence/add

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/add_licence

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            <xs:attribute name="licence" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="ok">
          <xs:complexType/>
        </xs:element>
        <xs:element name="cancel">
          <xs:complexType/>
        </xs:element>
        <xs:element name="error" type="errorType" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Примеры:

1. Пример успешного добавления лицензии.

Запрос:

http://192.168.1.21:9999/commands/add_licence

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="add_licence.xsd">
  <request storage="ds1" licence="0496c0cf9da7bb3b119f7557324de88c50d5e93276748e4e8717898426700
bdb454bdf5a280b8b3d2ac880cbfef55e342fa2cc43e1cad53a7e60fbde7536058ae6d327f0bbb9f48dd899e427868f1a
7d9313ba739c00a1f64beb0e40166ceae31deabbba6ec3860ebc99c253397fbadb24d0b25437b34846b4575ad1f6561a
8-5ce28b51f87cb09852d0c2c922bdf64f44fe03368480ef99c232f5265560d2142ec9e714c04f9f5e64e713b450b380a
6a6e82f3d5ae9a29299a3ed752f3052610cdc66a318d0fb67d3ac77e0bcc07af5aa48b3a270a371a818d8a30ef14d6999
db7a2490df4065a647c06951a6c51f11081a27757e28920677b8964428e41289-
f5ae26d9fc02611e1ad7f44f0b76303c907844e32ffb24ddea117f9f16126f8442c40d1155db65b429bfb1fe84c76f5b2
b9d6fffd3550da890245f0811992300739552f10f8e5e9c745a67646e51ad61174bd3980b28f73ee406e667e70cf937f3c
42b6eeebdb4665f33f64506bfe19ce324214cf8ee74f9b45b9ed2e9627077a-
e9e6b278b677963cf21f074dc091d05cdaa65888f73034d9d961a77da1fb824bf650c152119570aa1ff4c081815546141
64d0d3361505d27dbc423d5f62523c6dc42a1c77003313606df5004e7664d26be23dd9f407f8a527c1b9f38d5363ffb3
233b46444b05da0786c30baa480d847ccbb757685a861d1499724d5108e02a-91500ed1a7da43b16ce8b9595eeda66788
30309b64e33e718a594d0b6cd079f38ebcba10dac2bf7e28fed6d6c6fe3df6e1cbd36467163139d5b1e014f2312f0d761
9ad9a27a1b2714b3c40164bdb750481c5bfc1399006391e4282d5e8ad9e8c6a334b0a32b5edaeafc42bd2a2ccaf1be99c9
68aa376b11ec48b00292ece8c4ac-062ca6ef60cb14cb31364f5aa15edf9b29f3fd79cfb773c5ed528e104bcc2115b7a9
ae70d79240a7f5c26607b61b347b9ee69ce874250f0a796be589da1948c80a709bf0bc68fa5e274dd1689ad2c46b117de
fb438d86fb71a5844fec124ddd3d3b3cdd83d188243cd7e5f3daa32c023d169fb0e65d7c9ec4bb595904ed3ac1e-
a715d496e9d7e895df0b88a24a8ecd747234681d8788165b79091934b4ba009c3c230457fc50f16f533a3d7bda8bee7be
b7ede7b08a9ba7dc5c7b59ecfbf1f9c2f43a659f31450d15a6132006e8a23b3fe7bd84ac992eb94e613d9107fb60f60f1
a3f3d382e966da31b04b57fd00394f4221fc0d98e21a6c060b097798ca1359-5a23735be4ee3d2779d633b9d0f0d09f5b
88fb9fc0ab80424c778d34d2c9dacc9d9577edb2196a56c31961a8f62eca2d7ea92746b0e266d4313a30a2b5f751c7be3
feb11fc7601e7103748ecffaa01a09aab364f9182d862dfbbc4b14c79147d06b380cd77f7ab26f130b38553027d03b0ac
aa89a3e061eeb889d229160b823a-8ab6f5fb350ee901ed127f2591d69e4bb7007673b1e46421321a5355c7b9d8eca8f1
ffcfbb09c73cd2ae2a37912159ae58b77d9a006ec5784726b881a5afc45f1ffbdb51fab7bd83fd874d37df8a68032763a
078bfee1dbe8588e403b7e0c1f2cc2cabd4ee130977af342614e1b4a0630acead15ec28d191eddf51b4596bb1af-887ea
4cab7ed20f75ed68f2e7fc169d85601365d600c102fddb73436992de4b68d09c687c1ff2ff68809009d3ab6f1b2a4dc6
52c6eade0fd317dfa42eea0bc65335c4f6459ff4cf55ab9b497cdc9e71e1977287187c0c2c97f7e8ca44be3115c573ba
fe6814f3f2306a2b7f8fb1b8f059ee49ffceb01cecc9cab5b61abd0282"/>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="add_licence.xsd">
  <ok/>
</out>

```

2. Пример неуспешного добавления лицензии.

Запрос: http://192.168.1.21:9999/commands/add_licence

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="add_licence.xsd">
  <request storage="ds1" licence="bad-licence-11111111qwertyqwertyqwertyqwerty111111111"/>
</in>
```

Код ответа:

404

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="add_licen
ce.xsd">
  <error cmd="add_licence" reason="bad_licence"/>
</out>
```

Ис_current_limits

Команда позволяет просмотреть текущие лицензионные ограничения ECSS-10.

Аналог команды в CoCon:

cluster/storage/<CLUSTER>/licence/current-limits

где

<CLUSTER> - имя кластера хранения долговременных данных (DS).

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/current_limits

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

list_licence.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />
  <!-- propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- licenceType-->
  <xs:complexType name="licenceType">
    <xs:sequence>
      <xs:element name="limits">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="
unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="infos">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="
unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="id" type="xs:integer" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
    <xs:attribute name="hex" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>

```



```

<xs:element name="licences">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="licence" type="licenceType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Ответ включает в себя "одну лицензию" с ID = "-1" и содержит только текущие лицензионные ограничения.

Пример:

Запрос: http://192.168.1.21:9999/commands/current_limits

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="list_licence.xsd">
  <request storage="ds1" />
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="list_licence.xsd">
  <licences>
    <licence id="-1" active="true" hex="">
      <limits>
        <property key="Channels on dialer outgoing calls" value="10"/>
        <property key="IVR script's customization enabled (obsolete)" value="true"/>
        <property key="Total count of simultaneous records voice calls" value="200"/>
        <property key="The mode of 'TSMN backup trunk' work" value="backup"/>
        <property key="Concurrent calls on TSMN system (backup)" value="4"/>
        <property key="Concurrent calls on TSMN system (active)" value="5"/>
        <property key="Active teleconferences" value="26"/>
        <property key="Max members of one teleconference" value="199"/>
        <property key="Teleconference channels" value="200"/>
        <property key="Call center active supervisors" value="10"/>
        <property key="Call center active agents" value="100"/>
        <property key="Call center operator's capability for look at calls in queue"
value="true"/>
        <property key="Call center operator's capability for use Intervension SS"
value="true"/>
        <property key="CDR generation mode(s)" value="basic"/>
        <property key="Sorm enabled" value="true"/>
        <property key="Channels on SORM mediator" value="20"/>
        <property key="Total count of members of the one Meet Me conference" value="40"/>
        <property key="Total count of the Meet Me conferences" value="20"/>
        <property key="Total count of members of the one chatroom" value="40"/>
        <property key="Total count of chat rooms" value="20"/>
        <property key="Total count of members of the one add-on conference" value="30"/>
        <property key="Total count of Add-on conferences" value="10"/>
        <property key="Virtual subscribers limit" value="infinity"/>
        <property key="Subscribers limit" value="2000"/>
        <property key="System alive time without LPM token (min)" value="1d"/>
        <property key="Cluster md1 (mediator) nodes" value="md1@ecss1,md1@ecss2"/>
        <property key="Cluster sip1 (adapter) type" value="sip"/>
        <property key="Cluster sip1 (adapter) nodes" value="sip1@ecss1,sip1@ecss2"/>
        <property key="Cluster megaco1 (adapter) type" value="megaco"/>
        <property key="Cluster megaco1 (adapter) nodes"
value="megaco1@ecss1,megaco1@ecss2"/>
        <property key="Cluster core1 (core) nodes" value="core1@ecss1,core1@ecss2"/>
        <property key="Cluster core1 (core) redundancy" value="true"/>
        <property key="Cluster core1 (core) max call attempts" value="2000"/>
        <property key="Cluster core1 (core) max conversation time, sec" value="86400"/>
        <property key="Cluster ds1 (storage) nodes" value="ds1@ecss1,ds1@ecss2"/>
        <property key="Cluster bus1 (bus) nodes" value="mycelium1@ecss1,mycelium1@ecss2"/>
      >
    </limits>
  </infos/>

```

```
</licence>  
</licences>  
</out>
```

Ис_list_licence

Команда позволяет просмотреть список существующих лицензий в базе данных ECSS-10.

Аналог команды в CoCon:

cluster/storage/<CLUSTER>/licence/list-licence

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/list_licence

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />
  <!-- propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- licenceType-->
  <xs:complexType name="licenceType">
    <xs:sequence>
      <xs:element name="limits">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="infos">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="id" type="xs:integer" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
    <xs:attribute name="hex" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>

```

```

    <xs:element name="licences">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="licence" type="licenceType" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="error" type="errorType"/>
  </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Каждая лицензия включает в себя ограничения (limits) и информацию об администраторе, который сгенерировал лицензию.

Возможны следующие ограничения:

Таблица 1. Описание общих параметров лицензии

| Параметр | Описание |
|----------------------------|--|
| SSW ID | Идентификатор программного коммутатора SSW |
| Description | Описание владельца лицензии |
| Creation date | Дата создания лицензии |
| Expiration date (UTC) | Дата окончания срока действия лицензии |
| Author | Автор лицензии |
| Author IP | Сетевое имя системы автора лицензии |
| | Сетевые адреса системы автора лицензии |
| Encoder user | Логин, используемый для генерации лицензии |
| Encoder host | Сетевое имя системы для генерации лицензии |
| Encoder IP | Сетевые адреса системы для генерации лицензии |
| Active | Активность лицензии |
| Cluster bus | Описание кластера Bus |
| name | Имя кластера Bus |
| nodes | Список нод кластера Bus |
| Cluster storage | Описание кластера Storage |
| name | Имя кластера Storage |
| nodes | Список нод кластера Storage |
| Cluster core | Описание кластера Core |
| name | Имя кластера Core |
| nodes | Список нод кластера Core |
| redundancy | Поддержка резервирования |
| max call attempts | Максимальное количество одновременных соединений |
| max conversation time, sec | Максимальная длительность вызова |
| Cluster adapter | Описание кластера адаптера |
| name | Имя кластера адаптера |
| type | Тип кластера адаптера |
| nodes | Список нод кластера адаптера |
| Cluster mediator | Описание кластера Mediator |
| name | Имя кластера Mediator |

| Параметр | Описание |
|-------------------------|--|
| nodes | Список нод кластера Mediator |
| Cluster virtual name | Описание виртуального кластера Bridge Имя виртуального кластера |
| Cluster virtual name | Описание виртуального кластера Имя виртуального кластера |

Таблица 2. Описание лицензионных ограничений

| Параметр | Описание |
|--|---|
| TSMN system | Параметры транковой услуги TSMN |
| concurrent calls (active) | Количество одновременных вызовов, отправляющих запрос на сервис TSMN по основному направлению |
| concurrent calls (backup) | Количество одновременных вызовов, отправляющих запрос на сервис TSMN по резервному направлению |
| backup mode | Режим резервирования TSMN транка. Возможные значения: backup – вызовы на второй транк идут только тогда, когда недоступен первый; load_share – вызовы на оба транка идут одновременно; none – резервирование транка выключено |
| Teleconference | Параметры услуги селекторной связи |
| channels | Количество участников всех конференций селекторной связи |
| max members of one teleconference | Максимальное количество участников одной конференции селекторной связи |
| active count | Максимальное количество активных конференций селекторной связи |
| SORM | Параметры COPM |
| enabled | Поддержка COPM (параметр устарел, вместо него теперь используется [system, sorm, channels]) |
| channels on SORM mediator | Количество подключений COPM-посредников к доменам ECSS-10 (например, если один COPM-посредник хочет COPMировать два домена, то [system, sorm, channels] должен быть равен 2 |
| MeetMe | Параметры MeetMe-конференции |
| total count | Максимальное количество участников в одной MeetMe-конференции |
| members of the one Meet Me conference | Максимальное количество одновременных MeetMe-конференций в системе |
| IVR | Параметры IVR |
| IVR script's customization enabled (obsolete) | Возможность ограничивать количество блоков в одном IVR-скрипте, а также список блоков для домена. (По умолчанию в домене не ограничено количество блоков в IVR-скрипте; все блоки доступны) |
| Add-on conferences | Ограничения конференций add-on в системе |
| total count | Максимальное количество одновременных конференций add-on в системе |
| Chat rooms | Ограничения конференц-комнат |
| total count | Максимальное количество одновременных chatroom-ов в системе |
| members of the one chatroom | Максимальное количество участников в одном chatroom-е |
| Call center | Параметры Call-центра |
| operator's capability for look at calls in queue | Возможность просматривать очередь Call-центра |
| operator's capability for use Intervension SS | Поддержка ДВО "Вмешательство" на операторе Call-центра |
| active agents | Количество операторов Call-центра |
| active supervisors | Количество супервизоров (старших операторов) Call-центра |
| CDR generation mode(s) | Режим генерации CDR |
| Virtual subscribers limit | Максимальное количество виртуальных абонентов |
| Subscribers limit | Максимальное количество абонентов |
| System alive time without LPM token (min) | Время, в течение которого ECSS-10 будет работать при отсутствии LPM-токена |

Набор параметров может различаться.

Пример:

Запрос:

http://192.168.1.21:9999/commands/list_licence

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="list_licence.xsd">
  <request storage="ds1" />
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="list_licence.xsd">
  <licences>
    <licence id="0" active="true"
hex="0d7ec7c2a3fdcd9135193913e404fb5dd2034112b64c3c6f3f059c9efc0d33ce0010e7c08be32967fd9fd
d9512176a2857a4d5111ee06e888ebec697224f329130fe7103c2193ecc42ebb3bee8a63b90513d58b7bae589e
38e35152775b29ab9ebc36d74968ad1a32e1368228388e6851f61129dc1ee864b331825a35eee4583-7653e6ea
1a26a120f976de575af99fa31f1a5038e40e8e788468282a87051cfc9db6610b181badd3f98107f4b1ffaf2474
efac41021236c719f50a28de960c1656c292b4714a7cd7125686776ae4ab928b40411a23d1a191eb89eb7fa3e3
206f6c7bac7fe45a3c2d6b00d109f58f1b034d4e3694c662410811f994538b4542-75162f4204c2fedb8b103
8c6ceaa3ace605f30fa0bda1d4e75d079213bbdd29659affa8859e314ee1e407740aa51abc6612094b65d8cc5c
7df0bbbe1f82db4b8d7564ff7f1d35d159ac07631516cbee52960197d0f962e7aeceb8bd045fb9aeced2ab1229
81bf7b2f2507318491f8788a9ad7c31853ba1774f6f3cd29611980f-52d8ac5d8f21b3baaa13d85fb05e0d7757
3bb0b18e2c8186f40b0c0687c88dbe65b1faec7c1f686d8e975c83908cbbfae09afabf37bed1c1fc0e0a6e182d
e0bd2e239fd8006a9d7cbaa8a9f9e5deefacc4ffdc174d838e76a641d2154ff30bbff36341aec218f87e756098
e20576486ee354e54f06cce8639a2d8d948e3a032a-4e699967683c396ba2d34276e83818c749fe386199d0e2d
3fbf52467e3c4cfbea85edf4006ae50fab0a1bc5127f8df75b123595c597d509dce22f43773ff2957e85bb773b
0aa375ee07f27e036f73328d703c94f76721b6a3be9f3babd20246960abf0aad4a5d0a4dba71437bfe4dc4c0ad
5237a2caa8dac6e1efd2ddc31595d-9f9c5e6dff47095eaa14feff59ab2a11f7b376c70474f064d5f1d9269938
401883831132554f47e5a063ffdefac0bc23e6420b4f82c3ab78fbc01e41b0c84b6df4452a23e32c33ce56be1d
13b79300baf4aec2ba7ec1e90f5d17301eb550b632db3455d46e0f126ac550e6d9448c221cd5d0855e49ff8501
f3dc08e4866fa404">
    <limits>
      <property key="Channels on dialer outgoing calls" value="2"/>
      <property key="IVR script's customization enabled (obsolete)"
value="false"/>
      <property key="Total count of simultaneous records voice calls" value="0"/
>
      <property key="The mode of 'TSMN backup trunk' work" value="none"/>
      <property key="Concurrent calls on TSMN system (backup)" value="0"/>
      <property key="Concurrent calls on TSMN system (active)" value="0"/>
      <property key="Active teleconferences" value="0"/>
      <property key="Max members of one teleconference" value="0"/>
      <property key="Teleconference channels" value="3"/>
      <property key="Call center active supervisors" value="0"/>
      <property key="Call center active agents" value="0"/>
      <property key="CDR generation mode(s)" value="basic"/>
      <property key="Sorm enabled" value="false"/>
```



```

value="0"/>
    <property key="Total count of members of the one Meet Me conference"
value="16"/>
    <property key="Total count of the Meet Me conferences" value="0"/>
    <property key="Total count of members of the one chatroom" value="16"/>
    <property key="Total count of chat rooms" value="30"/>
    <property key="Total count of members of the one add-on conference"
value="16"/>
    <property key="Total count of Add-on conferences" value="30"/>
    <property key="Virtual subscribers limit" value="infinity"/>
    <property key="Subscribers limit" value="10"/>
    <property key="System alive time without LPM token (min)" value="1d"/>
    <property key="Cluster md1 (mediator) nodes" value="md1@ecss1"/>
    <property key="Cluster sip1 (adapter) type" value="sip"/>
    <property key="Cluster sip1 (adapter) nodes" value="sip1@ecss1"/>
    <property key="Cluster megaco1 (adapter) type" value="megaco"/>
    <property key="Cluster megaco1 (adapter) nodes" value="megaco1@ecss1"/>
    <property key="Cluster core1 (core) nodes" value="core1@ecss1"/>
    <property key="Cluster core1 (core) redundancy" value="false"/>
    <property key="Cluster core1 (core) max call attempts" value="5"/>
    <property key="Cluster core1 (core) max conversation time, sec"
value="60"/>
    <property key="Cluster ds1 (storage) nodes" value="ds1@ecss1"/>
    <property key="Cluster bus1 (bus) nodes" value="bus1@ecss1"/>
</limits>
<infos>
    <property key="Expiration date (UTC)" value="no limit"/>
    <property key="Description" value="Default licence"/>
</infos>
</licence>
<licence id="2" active="true"
hex="0496c0cf9da7bb3b119f7557324de88c50d5e93276748e4e8717898426700bdb454bdfea280b8b3d2ac88
0cbfef55e342fa2cc43e1cad53a7e60fbde7536058ae6d327f0bbb9f48dd899e427868f1a7d9313ba739c00a1f
64beb0e40166ceae31deabba6ec3860ebc99c253397fbadb24d0b25437b34846b4575ad1f6561a8-5ce28b51
f87cb09852d0c2c922bdf64f44fe03368480ef99c232f5265560d2142ec9e714c04f9f5e64e713b450b380a6a6
e82f3d5ae9a29299a3ed752f3052610cdc66a318d0fb67d3ac77e0bcc07af5aa48b3a270a371a818d8a30ef14d
6999db7a2490df4065a647c06951a6c51f11081a27757e28920677b8964428e41289-
f5ae26d9fc02611e1ad7f44f0b76303c907844e32fffb24ddea117f9f16126f8442c40d1155db65b429bfb1fe84
c76f5b2b9d6ffd3550da890245f0811992300739552f10f8e5e9c745a67646e51ad61174bd3980b28f73ee406e
667e70cf937f3c42b6eeebdb4665f33f64506bfe19ce324214cf8ee74f9b45b9ed2e9627077a-
e9e6b278b677963cf21f074dc091d05cdaa65888f73034d9d961a77da1fb824bf650c152119570aa1ff4c08181
554614164d0d3361505d27dbcb423d5f62523c6dc42a1c77003313606df5004e7664d26be23dd9f407f8a527c1
b9f38d5363ffb3233b46444b05da0786c30baa480d847ccbb757685a861d1499724d5108e02a-91500ed1a7da4
3b16ce8b9595eeda6678830309b64e33e718a594d0b6cd079f38ebc10dac2bf7e28fed6d6c6fe3df6e1cbd36
467163139d5b1e014f2312f0d7619ad9a27a1b2714b3c40164bdb750481c5bfc1399006391e4282d5e8ad9e8c6
a334b0a32b5edaefc42bd2a2ccaf1be99c968aa376b11ec48b00292ece8c4ac-062ca6ef60cb14cb31364f5aa1
5edf9b29f3fd79cfb773c5ed528e104bcc2115b7a9ae70d79240a7f5c26607b61b347b9ee69ce874250f0a796b
e589da1948c80a709bf0bc68fa5e274dd1689ad2c46b117defb438d86fb71a5844fec124ddd3d3b3cdd83d1882
43cd7e5f3daa32c023d169fb0e65d7c9ec4bb595904ed3ac1e-
a715d496e9d7e895df0b88a24a8ecd747234681d8788165b79091934b4ba009c3c230457fc50f16f533a3d7bda
8bee7beb7ede7b08a9ba7dc5c7b59ecfbf1f9c2f43a659f31450d15a6132006e8a23b3fe7bd84ac992eb94e613
d9107fb60f60f1a3f3d382e966da31b04b57fd00394f4221fc0d98e21a6c060b097798ca1359-5a23735be4ee3
d2779d633b9d0f0d09f5b88fb9fc0ab80424c778d34d2c9dacc9d9577edb2196a56c31961a8f62eca2d7ea9274
6b0e266d4313a30a2b5f751c7be3feb11fc7601e7103748ecffaa01a09aab364f9182d862dfbbc4b14c79147d0
6b380cd77f7ab26f130b38553027d03b0acaa89a3e061eeb889d229160b823a-8ab6f5fb350ee901ed127f2591
d69e4bb7007673b1e46421321a5355c7b9d8eca8f1ffcbb09c73cd2ae2a37912159ae58b77d9a006ec5784726
b881a5afc45f1ffbbdb51fab7bd83fd874d37df8a68032763a078bfee1dbe8588e403b7e0c1f2cc2cabd4ee1309
77af342614e1b4a0630acead15ec28d191eddf51b4596bb1af-887ea4cab7ed20f75ed68f2e7fc169d8560136
5d600c102fd73436992de4b68d09c687c1ff2ff68809009d3ab6f1b2a4dc652c6eade0fd317dfafa42eea0bc6
5335c4f6459ff4cf55ab9b497cdc9e71e1977287187c0c2c97f7e8ca44be3115c573baf6814f3f2306a2b7f8f

```

```

b1b8f059ee49ffceb01cec9cab5b61abd0282">
  <limits>
    <property key="Cluster bus1 (bus) nodes"
value="mycelium1@ecss1,mycelium1@ecss2"/>
    <property key="Cluster ds1 (storage) nodes" value="ds1@ecss1,ds1@ecss2"/>
    <property key="Cluster core1 (core) nodes"
value="core1@ecss1,core1@ecss2"/>
    <property key="Cluster core1 (core) redundancy" value="true"/>
    <property key="Cluster core1 (core) max call attempts" value="2000"/>
    <property key="Cluster core1 (core) max conversation time, sec"
value="86400"/>
    <property key="Cluster sip1 (adapter) type" value="sip"/>
    <property key="Cluster sip1 (adapter) nodes"
value="sip1@ecss1,sip1@ecss2"/>
    <property key="Cluster md1 (mediator) nodes" value="md1@ecss1,md1@ecss2"/>
    <property key="Subscribers limit" value="2000"/>
    <property key="Total count of Add-on conferences" value="10"/>
    <property key="Total count of members of the one add-on conference"
value="30"/>
    <property key="Total count of chat rooms" value="20"/>
    <property key="Total count of members of the one chatroom" value="40"/>
    <property key="Sorm enabled" value="true"/>
    <property key="Channels on SORM mediator" value="20"/>
    <property key="Call center operator's capability for look at calls in
queue" value="true"/>
    <property key="Call center operator's capability for use Intervension SS"
value="true"/>
    <property key="Call center active agents" value="100"/>
    <property key="Call center active supervisors" value="10"/>
    <property key="Teleconference channels" value="200"/>
    <property key="Max members of one teleconference" value="199"/>
    <property key="Active teleconferences" value="26"/>
    <property key="Concurrent calls on TSMN system (active)" value="5"/>
    <property key="Concurrent calls on TSMN system (backup)" value="4"/>
    <property key="The mode of 'TSMN backup trunk' work" value="backup"/>
    <property key="Total count of simultaneous records voice calls"
value="200"/>
    <property key="IVR script's customization enabled (obsolete)"
value="true"/>
    <property key="Channels on dialer outgoing calls" value="10"/>
  </limits>
  <infos>
    <property key="Encoder IP" value="172.16.0.51, 127.0.0.1"/>
    <property key="Encoder host" value="droid"/>
    <property key="Encoder user" value="ssw"/>
    <property key="Author IP" value="172.16.0.51, 127.0.0.1"/>
    <property key="Author" value="ssw"/>
    <property key="Expiration date (UTC)" value="31.12.2019 23:59:59"/>
    <property key="Creation date" value="02.04.2019 15:45:00"/>
    <property key="Description" value="Biysk Test"/>
    <property key="SSW ID" value="ECSS 010079"/>
  </infos>
  </licence>
  <licence id="3" active="true"
hex="09e595f120c9d375eb45f27ebb33a3fbb6ea8702118e93b1818edb725fe70208caa668e931082f9aca0dc
212e7a2187f44fac22d0082f0eaf7d8db01dd69b51ba3864ed9db942d4dee8f3de17e298386fc1d2e4046fd57d
0b6c27753c3a5f680d1b3fb62388b0c88de3c69d103b3a0c359b85537b363cbfd53a135f6a08b0792-25cd49bd
8c5657e7702c770975e2e6f46e7b1484012d5e50f39fc36c452ee222551b684474d503c673edd78a50c796d6
8cd0d06f1ad3ef2f93528fe0e5f473a22082ea6d6bf9ea8cfc570c69ce164cb2fcb233f25ece3119fc4d093e37
d34bf3e6ec80a356ef34ecb0f6d408d72d031f29d2e97b06511879862573f94dd65f-1ef6ec6fc0b7fc996a694

```

```

f70267b2e43cd6d6acd2d1d7b8a9201d54fda4903279d56aa00169d5422bdde8cd74b5621fc8021475f32dbaca
65d1184f3c0803eb14b9ee15563ca076e78c0548ca9c72f324e2626ea53648cc4017ca113ea522cc7523c125b9
c07ca99196f40f107a806b4e82ce122fa773e53379894722458355e-
f025b36b83a71fd327da3642b95a70cc0e42fac469ab3a610e2b0a567df2f2140b7a88ed9d9280cb142374c350
f461207f785e73869bd9377d62a3080f832c0da4aad5c653d6e5f8653176779245a6957ab15698f4606f0d8b16
22e7032e6db8e9c326bc2c86740cd4e28b5850995782d584fc26fe0cdaca81cd80dd2d1d289f-98ab44b7fcc60
b5acffe13cde7f1364a91bb263959581c1de6be29856dccff2217be7f700c66a25d454f7f8ca170298877cec52
0d08856ba15d09124162034d9d9951da63bc1e9c9f636bd72f88e7cb74e319639d2962262b912bab53691edb83
e7d9121881a776ab580d57c99af0339ff70140b09e3f2ed3d23fee08ff0f114-15d16c980b7422e1b14168cda7
6186108ca3ed8983befe38d3674f32d0eaefce896fb05186107c2334e114ece629530bac377fe5dfd37347113c
4b0de7caa84cdd16b29aae4e627fcf0b54c38502dabed553500da70d5a777b9d891f059a96cd4fda388bea1df9
4bc3187a6355e718ac88549b3bc6d2438f4c8a20dbb0c9e190-2620129bf693bb3831ef74ac382d14428d195ef
de44b3980d5852a3ad34daef7e76b8cd10d9fec46755bc412fc959bef4954267c23b3d6fea117c77389e558ff0
8f6b0a82d0b412a26bf681f2ae3a888169ca7492fe99761b4494d7ffecd4e46c95fa64f69a29a6e972032f50c4
9428b2494c778bd3e976b9dd455543dde7278-449ac65fe58f26949e5d5888e9400a40a2a951f5c67082dd4e8d
e616c1f6f1980a0bbabf6572063eceb60b7df28c9829d996d2fe3de760f364d9c454ea60d6724452e4066a6dae
9c9452160e773ab3fb2b06963d525e064bf6a4d6fdaca6e5cc8d0f1209cc7625d6366efbe160f6323edfd2b62b
f4a79265caf1e9cb3e7a307c-640298e0834b2572fe0cc1b6358565150f8a45b96c034aab66e5a4269b0ee9f4b
3713d75d4556d3446877952e3b9457108b24b967f35fbd5afd9aeafb583d901743ae97c39c121187297d5fa50a
51a56d4820b896a48fc9a063cec9d1ad57dd77db170d49374e682c105d24f11c5f868bf690a5eada939c4e5cb3
9ccafa7b05a-005f2326df40e72da954748f73e36c1223bf744e5310579256fce54ae3df88accf9f61d5960ded
55cb5b8075c10265bcd0fba2186308a2a1c37dc233a16d398416819f073515e831bfcf308359d870e0969f81e8
dcb4739ae5023aedb1bd0c21b4ecf5e90961ca482420e18203d7edb52ce10bdc9fe3da268085bc47433d00bf">
  <limits>
    <property key="Cluster bus1 (bus) nodes"
value="mycelium1@ecss1,mycelium1@ecss2"/>
    <property key="Cluster ds1 (storage) nodes" value="ds1@ecss1,ds1@ecss2"/>
    <property key="Cluster core1 (core) nodes"
value="core1@ecss1,core1@ecss2"/>
    <property key="Cluster core1 (core) redundancy" value="true"/>
    <property key="Cluster core1 (core) max call attempts" value="2000"/>
    <property key="Cluster core1 (core) max conversation time, sec"
value="86400"/>
    <property key="Cluster sip1 (adapter) type" value="sip"/>
    <property key="Cluster sip1 (adapter) nodes"
value="sip1@ecss1,sip1@ecss2"/>
    <property key="Cluster megaco1 (adapter) type" value="megaco"/>
    <property key="Cluster megaco1 (adapter) nodes"
value="megaco1@ecss1,megaco1@ecss2"/>
    <property key="Cluster md1 (mediator) nodes" value="md1@ecss1,md1@ecss2"/>
    <property key="Subscribers limit" value="2000"/>
    <property key="Total count of Add-on conferences" value="10"/>
    <property key="Total count of members of the one add-on conference"
value="30"/>
    <property key="Total count of chat rooms" value="20"/>
    <property key="Total count of members of the one chatroom" value="40"/>
    <property key="Total count of the Meet Me conferences" value="20"/>
    <property key="Total count of members of the one Meet Me conference"
value="40"/>
    <property key="Sorm enabled" value="true"/>
    <property key="Channels on SORM mediator" value="20"/>
    <property key="Call center operator's capability for look at calls in
queue" value="true"/>
    <property key="Call center operator's capability for use Intervension SS"
value="true"/>
    <property key="Call center active agents" value="100"/>
    <property key="Call center active supervisors" value="10"/>
    <property key="Teleconference channels" value="200"/>
    <property key="Max members of one teleconference" value="199"/>
    <property key="Active teleconferences" value="26"/>

```

```

        <property key="Concurrent calls on TSMN system (active)" value="5"/>
        <property key="Concurrent calls on TSMN system (backup)" value="4"/>
        <property key="The mode of 'TSMN backup trunk' work" value="backup"/>
        <property key="Total count of simultaneous records voice calls"
value="200"/>
        <property key="IVR script's customization enabled (obsolete)"
value="true"/>
        <property key="Channels on dialer outgoing calls" value="10"/>
    </limits>
    <infos>
        <property key="Encoder IP" value="172.16.0.51, 127.0.0.1"/>
        <property key="Encoder host" value="droid"/>
        <property key="Encoder user" value="ssw"/>
        <property key="Author IP" value="172.16.0.51, 127.0.0.1"/>
        <property key="Author" value="ssw"/>
        <property key="Expiration date (UTC)" value="31.12.2020 23:59:59"/>
        <property key="Creation date" value="20.09.2019 06:51:00"/>
        <property key="Description" value="Biysk Test"/>
        <property key="SSW ID" value="ECSS 010079"/>
    </infos>
</licence>
</licences>
</out>

```

Ис_delete_licence

Команда удаляет лицензию из базы данных ECSS-10.

Аналог команды в CoCon:

cluster/storage/<CLUSTER>/licence/delete

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/delete_licence

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            <xs:attribute name="licence_id" type="xs:integer" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="ok">
          <xs:complexType/>
        </xs:element>
        <xs:element name="default_licence">
          <xs:complexType/>
        </xs:element>
        <xs:element name="not_found">
          <xs:complexType/>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema

```

Пример:

1. Пример успешного удаления лицензии.

Запрос: http://192.168.1.21:9999/commands/delete_licence

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_list_
of_clusters.xsd">
  <request storage="ds1" licence_id="2"/>
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="delete_licence.xsd">
  <ok/>
</out>
```

2. Пример неуспешного удаления лицензии.

Запрос: http://192.168.1.21:9999/commands/delete_licence

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_list_
of_clusters.xsd">
  <request storage="ds1" licence_id="0"/>
</in>
```

Код ответа:

404

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="delete_li
cence.xsd">
  <default_licence />
</out>
```

hc_set_passport

Команда установки паспорта ECSS-10.

Аналог команды в CoCon:

/cluster/storage/<CLUSTER>/licence/set-passport

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/set_passport

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            <xs:attribute name="passport" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="ok">
          <xs:complexType/>
        </xs:element>
        <xs:element name="error" type="errorType" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema
```

Пример 1:

Успешное добавление паспорта.

Запрос:

http://192.168.1.21:9999/commands/set_passport

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="set_passport.xsd">
  <request storage="ds1" passport="ddf96ecce552efcee2cdc365f4e6a4bd8a53ec9e971acf07c934f8a26dd9eafe8f2ba4dd651a493d5c9a863ec5ae46cc62ad27db98cc47a0b29f33b958c8f5ed8dde34eb4c4febb9a58e1e0ec5d9f687a754daa0959a0a09855a575fea14e5b87b7c090ed6ab1b415e0f1167996c07f0b7ed7d4508bcabca79e382a7bad37088-8a8a870e066bf835f9ce91ff674b981fd474b1cc5c58cff78dc4ebbf0edba72a4d2e760cab93719cfd1426df8e6fec9c0573f9474b50a3d22ac13869cc16c98744d57ad845f34fb708ed81e7b4bc76a969905280d336679819bb191898eebeb1b99945aa74a17dbed1c239f62c3d543e650f9b48379af8bc3a1ea821a5267a63-3d3011f1296f43413fd0e5dfcdd01dd7968211db6bfbdb19cd899e06463fa298d564a7f0a4f6dc61c83f191c117e5d1ab7696e81600ea03ee5a455506ca8cba3190029600b19830d81a5fba30ad42ef21c90f27b1912fb8e3ca5699a0cefc107000a15b5d37ada58ce78192368c9d03ea9845ef24e23ff07bf7e450fc566b8486-7c7ba7df1e45701e86d79532adcb7e282e4177055550e82924ffd7ebc9df2e90c591d5282b22b71ed3b8b8156bf9c36fe537031734437c2c24577ab7405dfa825c8557e13f03fc400d08de239477b2cbb4bdb5c01de358cf4c26f9e29f32b22ea2f4d5b3e6e8d882358a8e300805982762e04d356c27f787949234fcf110660"/>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="set_passport.xsd">
  <ok/>
</out>
```

Пример 2:

Неуспешное добавление паспорта.

Запрос: http://192.168.1.21:9999/commands/set_passport

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="set_passport.xsd">
  <request storage="ds1" passport="bad-passport-ffffffffffff23423251235127486353ffff"/>
</in>
```

Код ответа:

404

Ответ:


```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="set_passport.xsd">
  <error cmd="set_passport" reason="bad_token_info"/>
</out>
```

Ис_show_passport

Команда позволяет просмотреть паспорт системы ECSS-10.

Аналог команды в CoCon:

/cluster/storage/<CLUSTER>/licence/show-passport

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/show_passport

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- propertyType-->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- passportType-->
  <xs:complexType name="passportType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="ssw_id" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="create_date" type="xs:string" use="optional"/>
    <xs:attribute name="author" type="xs:string" use="optional"/>
    <xs:attribute name="author_host" type="xs:string" use="optional"/>
    <xs:attribute name="author_ip" type="xs:string" use="optional"/>
    <xs:attribute name="hex" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="passport" type="passportType" minOccurs="0" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

В паспорте доступна следующая информация:

- SSW ID - уникальный идентификатор версии системы ECSS-10;
- Description - описание лицензии;
- Creation date - дата создания лицензии;
- Author - автор лицензии;
- Author host - имя хоста, на котором была сгенерирована лицензия;

- Author IP - список IP-адресов (через запятую) машины, на которой была сгенерирована лицензия;
- Hex - паспорт ECSS-10 в HEX-кодировке.

Каждый из вышеописанных параметров кроме «Hex» может как присутствовать, так и отсутствовать.

❗ В случае если паспорт не задан, возвращается пустой «passport».

Пример:

Запрос:

http://192.168.1.21:9999/commands/show_passport

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="show_passport.xsd">  
  <request storage="ds1" />  
</in>
```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="show_passport.xsd">
  <passport author_ip="172.16.0.51, 127.0.0.1" author="ssw" create_date="03.10.2014"
    description="Biysk Test" ssw_id="ECSS 010079"
    hex=
    "ddf96ecce552efcee2cdc365f4e6a4bd8a53ec9e971acf07c934f8a26dd9eafe8f2ba4dd651a493d5c9a863ec5ae46cc
    62ad27db98cc47a0b29f33b958c8f5ed8dde34eb4c4febb9a58e1e0ec5d9f687a754daa0959a0a09855a575fea14e5b87
    b7c090ed6ab1b415e0f1167996c07f0b7ed7d4508bcabca79e382a7bad37088-8a8a870e066bf835f9ce91ff674b981fd
    474b1cc5c58cff78dc4ebbf0edba72a4d2e760cab93719cfd1426df8e6fec9c0573f9474b50a3d22ac13869cc16c98744
    d57ad845f34fb708ed81e7b4bc76a969905280d336679819bb191898eebeb1b99945aa74a17dbed1c239f62c3d543e650
    f9b48379af8bc3a1ea821a5267a63-3d3011f1296f43413fd0e5dfcdd01dd7968211db6bfd19cd899e06463fa298d564
    a7f0a4f6dc61c83f191c117e5d1ab7696e81600ea03ee5a455506ca8cba3190029600b19830d81a5fba30ad42ef21c90f
    27b1912fb8e3ca5699a0cef107000a15b5d37ada58ce78192368c9d03ea9845ef24e23ff07bf7e450fc566b8486-7c7b
    a7df1e45701e86d79532adcb7e282e4177055550e82924ffd7ebc9df2e90c591d5282b22b71ed3b8b8156bf9c36fe5370
    31734437c2c24577ab7405dfa825c8557e13f03fc400d08de239477b2cbb4bdb5c01de358cf4c26f9e29f32b22ea2f4d
    5b3e6e8d882358a8e300805982762e04d356c27f787949234fcf11066f">
    <property key="encoder_create_date" value="{{2014, 10, 3}, {6, 2, 55}}"/>
    <property key="encoder_ip" value="[172, 16, 0, 51], [127, 0, 0, 1]"/>
    <property key="encoder_hostname" value=""droid""/>
    <property key="encoder_user" value=""ssw""/>
  </passport>
</out>

```

 12.11.2019

Нс_md_rpss - Управление свойствами кластера Mediator (RPSS)

- [Нс_md_rpss_info](#)
- [Нс_md_rpss_set](#)
- [Нс_md_rpss_clean](#)

Нс_md_rpss_info

Команда для просмотра информации о кластере Mediator (RPSS).

Аналог команды в CoCon:

cluster/mediator/<CLUSTER>/properties/rpss/info

Шаблон URL-запроса:

http://host:port/commands/md_rpss_info

Коды ответа:

- 200 - успех;
- 404 - ошибка.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="md_rpss_info">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get properties of Mediator cluster (RPSS).
      Usage: http://server:9999/commands/md_rpss_info
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

<!-- Types -->

<!-- dbConnection value -->
<!--<xs:simpleType name="dbConnectionsType">-->
  <!--<xs:restriction base="xs:string">-->
    <!--<xs:pattern value="(name=.;provider=[a-zA-Z]+;database=[a-zA-Z0-9$_]
{1,64};user=. {1,16};password=.;host=((( [0-9]{1,3}\.){3}[0-9]{1,3})|((( [a-zA-Z]| [a-zA-Z][a-zA-
Z0-9\ - ]*[a-zA-Z0-9])\ .)* ([A-Za-z]| [A-Za-z][A-Za-z0-9\ - ]*[A-Za-z0-9])));port=[0-9]{1,5}, ?)" />-->
    <!--</xs:restriction>-->
  <!--</xs:simpleType>-->

<!-- alarmPanelSignals value -->
<!--<xs:simpleType name="alarmPanelSignalsType">-->
  <!--<xs:restriction base="xs:string">-->
    <!--<xs:pattern value="((critical|major|warning|indeterminate):((( [0-9]{1,4}: [0-9]+:
[0-9]+: [0-9]+) | default));)" />-->
    <!--</xs:restriction>-->

<!-- alarmPanel value -->
<!--<xs:simpleType name="alarmPanelType">-->
  <!--<xs:restriction base="xs:string">-->
    <!--<xs:pattern value="((local:(any|((( [0-9]{1,3}\.){3}[0-9]{1,3})|((( [a-zA-Z]| [a-zA-
Z][a-zA-Z0-9\ - ]*[a-zA-Z0-9])\ .)* ([A-Za-z]| [A-Za-z][A-Za-z0-9\ - ]*[A-Za-z0-9]))):[0-9]{1,5}){1};
(remote:((( [0-9]{1,3}\.){3}[0-9]{1,3})|((( [a-zA-Z]| [a-zA-Z][a-zA-Z0-9\ - ]*[a-zA-Z0-9])\ .)* ([A-Za-
z]| [A-Za-z][A-Za-z0-9\ - ]*[A-Za-z0-9]))):[0-9]{1,5}){1})" />-->
    <!--</xs:restriction>-->
  <!--</xs:simpleType>-->

<!-- Options type -->
<xs:complexType name="mdRPSSOptionsType">
  <xs:complexContent>
    <xs:extension base="baseOptionsType">
      <xs:attribute name="alarmNotifyCacheSize" type="positiveIntegerType"
use="optional" />
      <xs:attribute name="alarmNotifyInterval" type="positiveIntegerType"
use="optional" />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

```

        <xs:attribute name="alarmPanel" type="xs:string" use="optional" />
        <xs:attribute name="alarmPanelSignals" type="xs:string" use="optional" />
        <xs:attribute name="collectInterval" type="positiveIntegerType" use="optional" />
        <xs:attribute name="collectProcPriority" type="procPriorityType" use="optional" /
    >
        <xs:attribute name="dbConnections" type="xs:string" use="optional" />
        <xs:attribute name="defaultAlarmRowLimit" type="positiveIntegerType"
use="optional" />
        <xs:attribute name="exportEmptyReports" type="xs:boolean" use="optional" />
        <xs:attribute name="exportProcPriority" type="procPriorityType" use="optional" />
        <xs:attribute name="parallelExportStatistics" type="xs:boolean" use="optional" />
        <xs:attribute name="snmpAgentEnable" type="xs:boolean" use="optional" />
        <xs:attribute name="statProcessingCount" type="positiveIntegerType"
use="optional" />
        <xs:attribute name="maxClearedAlarms" type="positiveIntegerType" use="optional" /
    >
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

<!-- Types -->

<!-- Root -->

<!-- in -->
<xs:element name="in">
    <xs:complexType>
        <xs:all minOccurs="1" maxOccurs="1">
            <xs:element name="request" minOccurs="1" maxOccurs="1">
                <xs:complexType>
                    <xs:all minOccurs="1" maxOccurs="1">
                        <xs:element name="nc_options" type="ncInfoInType" minOccurs="1"
                            maxOccurs="1" />
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:all>
    </xs:complexType>
</xs:element>
<!-- in -->

<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" minOccurs="1" maxOccurs="1">
                <xs:complexType>
                    <xs:all minOccurs="1" maxOccurs="1">
                        <xs:element name="nc_options" type="ncInfoOutType" minOccurs="1"
                            maxOccurs="1" />
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:choice>
    </xs:complexType>
</xs:element>

```

```

        </xs:all>
        </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
        </xs:choice>
    </xs:complexType>
</xs:element>
<!-- out -->

<!-- Root -->

</xs:schema>

```

Запрос значений всех доступных свойств:

http://192.168.1.21:9999/commands/md_rpss_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="md_rpss_info.xsd">
    <request>
        <nc_options cluster="md1" node="md1@ecss1" />
    </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="md_rpss_info.xsd">
    <result>
        <nc_options>
            <cluster name="md1">
                <options xs:type="mdRPSSOptionsType"/>
            </cluster>
            <node name="md1@ecss1">
                <options xs:type="mdRPSSOptionsType" statProcessingCount="5"
snmpAgentEnable="false" parallelExportStatistics="false" maxClearedAlarms="10000"
exportProcPriority="normal" exportEmptyReports="false" defaultAlarmRowLimit="25"
dbConnections=
"name=stat_db;provider=mysql;database=ecss_statistics;user=statistics;password=statistics;host=19
2.168.1.71;port=3306" collectProcPriority="normal" collectInterval="10" alarmPanelSignals=""
alarmPanel="" alarmNotifyInterval="10" alarmNotifyCacheSize="30"/>
            </node>
        </nc_options>
    </result>
</out>

```

Нс_md_rpss_set

Команда для изменения свойств кластера Mediator (RPSS)

Аналог команды в CoCon:

cluster/mediator/<CLUSTER>/properties/rpss/set

Шаблон URL-запроса:

http://server:9999/commands/md_rpss_set

Коды ответа:

- 200 - успех;
- 404 - ошибка.

XML-схема:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="md_rpss_set">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set properties of Mediator cluster (RPSS).
      Usage: http://server:9999/commands/md_rpss_set
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

<!-- Types -->

<!-- dbConnection value -->
<xs:simpleType name="dbConnectionsType">
  <xs:restriction base="xs:string">
    <xs:pattern value="(name=.;provider=[a-zA-Z]+;database=[a-zA-Z0-9$_]{1,64};user={1,16};password=.;host=((([0-9]{1,3}\.){3}[0-9]{1,3})|((([a-zA-Z]|[a-zA-Z][a-zA-Z0-9\-\ ]*[a-zA-Z0-9])\.)*) ([A-Za-z]|[A-Za-z][A-Za-z0-9\-\ ]*[A-Za-z0-9])));port=[0-9]{1,5},?)" />
  </xs:restriction>
</xs:simpleType>

<!-- alarmPanelSignals value -->
<xs:simpleType name="alarmPanelSignalsType">
  <xs:restriction base="xs:string">
    <xs:pattern value="((critical|major|warning|indeterminate):([0-9]{1,4}:[0-9]+:[0-9]+:[0-9]+)|default);?" />
  </xs:restriction>
</xs:simpleType>

<!-- alarmPanel value -->
<xs:simpleType name="alarmPanelType">
  <xs:restriction base="xs:string">
    <xs:pattern value="((local:(any|((([0-9]{1,3}\.){3}[0-9]{1,3})|((([a-zA-Z]|[a-zA-Z][a-zA-Z0-9\-\ ]*[a-zA-Z0-9])\.)*) ([A-Za-z]|[A-Za-z][A-Za-z0-9\-\ ]*[A-Za-z0-9]))):[0-9]{1,5}){1};(remote:(([0-9]{1,3}\.){3}[0-9]{1,3})|((([a-zA-Z]|[a-zA-Z][a-zA-Z0-9\-\ ]*[a-zA-Z0-9])\.)*) ([A-Za-z]|[A-Za-z][A-Za-z0-9\-\ ]*[A-Za-z0-9]))):[0-9]{1,5}){1})" />
  </xs:restriction>
</xs:simpleType>

<!-- Options type -->
<xs:complexType name="mdRPSSOptionsType">
  <xs:complexContent>
    <xs:extension base="baseOptionsType">
      <xs:attribute name="alarmNotifyCacheSize" type="positiveIntegerType"
use="optional" />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

```

        <xs:attribute name="alarmNotifyInterval" type="positiveIntegerType"
use="optional" />
        <xs:attribute name="alarmPanel" type="alarmPanelType" use="optional" />
        <xs:attribute name="alarmPanelSignals" type="alarmPanelSignalsType"
use="optional" />
        <xs:attribute name="collectInterval" type="positiveIntegerType" use="optional" />
        <xs:attribute name="collectProcPriority" type="procPriorityType" use="optional" /
>
        <xs:attribute name="dbConnections" type="dbConnectionsType" use="optional" />
        <xs:attribute name="defaultAlarmRowLimit" type="positiveIntegerType"
use="optional" />
        <xs:attribute name="exportEmptyReports" type="xs:boolean" use="optional" />
        <xs:attribute name="exportProcPriority" type="procPriorityType" use="optional" />
        <xs:attribute name="parallelExportStatistics" type="xs:boolean" use="optional" />
        <xs:attribute name="snmpAgentEnable" type="xs:boolean" use="optional" />
        <xs:attribute name="statProcessingCount" type="positiveIntegerType"
use="optional" />
        <xs:attribute name="maxClearedAlarms" type="positiveIntegerType" use="optional" /
>
    </xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- Types -->

<!-- Root -->

<!-- in -->
<xs:element name="in">
    <xs:complexType>
        <xs:all minOccurs="1" maxOccurs="1">
            <xs:element name="request" minOccurs="1" maxOccurs="1">
                <xs:complexType>
                    <xs:all minOccurs="1" maxOccurs="1">
                        <xs:element name="nc_options" type="ncSetInType" minOccurs="1"
maxOccurs="1" />
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:all>
    </xs:complexType>
</xs:element>

<!-- in -->

<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" minOccurs="1" maxOccurs="1">
                <xs:complexType>

```

```

        <xs:all minOccurs="1" maxOccurs="1">
            <xs:element name="nc_options" type="ncSetOutType" minOccurs="1"
                maxOccurs="1" />
        </xs:all>
    </xs:complexType>
</xs:element>
<xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
</xs:choice>
</xs:complexType>
</xs:element>
<!-- out -->

<!-- Root -->

</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/md_rpss_set

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="md_rpss_set.xsd">
  <request>
    <nc_options>
      <cluster name="md1">
        <options xs:type="mdRPSSOptionsType" alarmNotifyInterval="10" />
      </cluster>
      <node name="md1@ecss1">
        <options xs:type="mdRPSSOptionsType"
dbConnections=
"name=test;provider=mysql;database=md_rpss_test;user=ssw;password=ssw;host=192.168.1.71;port=3306
,name=test-1;provider=mysql;database=ssw;user=ssw;password=ssw;host=localhost;port=3306"
alarmPanelSignals="critical:default;warning:1000:3:10:10" alarmPanel="local:any:
1234;remote:localhost:6789" />
      </node>
    </nc_options>
  </request>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="md_rpss_set.xsd">
  <result>
    <nc_options>
      <cluster name="md1">
        <success option="alarmNotifyInterval"/>
      </cluster>
      <node name="md1@ecss1">
        <success option="alarmPanel"/>
        <success option="alarmPanelSignals"/>
        <success option="dbConnections"/>
      </node>
    </nc_options>
  </result>
</out>
```

Нс_md_rpss_clean

Командой удаляются установленные значения свойств кластера Mediator.

Аналог команды в CoCon:

cluster/mediator/<CLUSTER>/properties/rpss/clean

Шаблон URL-запроса:

http://server:9999/commands/md_rpss_clean

Коды ответа:

- 200 - успех;
- 404 - ошибка.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="md_rpss_clean">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean property or properties of RPSS.
      Usage: http://server:9999/commands/md\_rpss\_clean
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="nc_options" type="ncCleanInType" minOccurs="1"
                maxOccurs="1" />
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="nc_options" type="ncCleanOutType" minOccurs="1"
                maxOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- out -->

</xs:schema>

```

Примеры:Запрос: http://192.168.1.21:9999/commands/md_rpss_clean

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="md_rpss_clean.xsd">
  <request>
    <nc_options>
      <cluster name="md1">
        <option name="tts_realm" />
      </cluster>
      <node name="md1@ecss1">
        <option name="dbConnections" />
        <option name="alarmPanel" />
        <option name="alarmPanelSignals" />
      </node>
    </nc_options>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="md_rpss_clean.xsd">
  <result>
    <nc_options>
      <cluster name="md1">
        <success option="tts_realm"/>
      </cluster>
      <node name="md1@ecss1">
        <success option="alarmPanel"/>
        <success option="alarmPanelSignals"/>
        <success option="dbConnections"/>
      </node>
    </nc_options>
  </result>
</out>
```

Hc_media_profile - Управление медиа-профилями на виртуальной АТС

- [Hc_media_profile_list](#)
- [Hc_media_profile_declare](#)
- [Hc_media_profile_info](#)
- [Hc_media_profile_set](#)

- [Hc_media_profile_remove](#)

[Hc_media_profile_list](#)

Команда для просмотра списка медиа-профилей сконфигурированных в домене.

Аналог команды в CoCon:

domain/DOMAIN_NAME/media-profile/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/media_profile_list

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="media_profile.xsd"/>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok" type="okType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>

  <!-- okType -->
  <xs:complexType name="okType">
    <xs:sequence minOccurs="0" maxOccurs="unbounded">
      <xs:element name="media_profile" type="mediaProfileType"/>
    </xs:sequence>
  </xs:complexType>

  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

</xs:schema

```


Примеры:

http://192.168.1.21:9999/commands/media_profile_list

Запрос:

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_profile_list.xsd">
  <response>
    <ok>
      <media_profile domain="biysk.local" name="default" description="Default media profile
for current domain." codecs_priority="user">
        <codecs_info type="<other>">
          <codecs name="<other>" payload_type="*" enabled="true"/>
          <properties key="offroad" value="false"/>
          <properties key="rtcp-enabled" value="true"/>
        </codecs_info>
        <codecs_info type="audio">
          <codecs name="G722" payload_type="*" enabled="true"/>
          <codecs name="PCMA" payload_type="*" enabled="true"/>
          <codecs name="PCMU" payload_type="*" enabled="true"/>
          <codecs name="G729" payload_type="*" enabled="true"/>
          <codecs name="telephone-event" payload_type="*" enabled="true"/>
          <properties key="offroad" value="false"/>
          <properties key="rtcp-enabled" value="true"/>
        </codecs_info>
      </media_profile>
      <media_profile domain="biysk.local" name="mp1" description="add_video"
codecs_priority="system">
        <codecs_info type="video">
          <codecs name="H264" payload_type="*" enabled="true">
            <attributes key="key3" value="val3"/>
          </codecs>
          <properties key="offroad" value="true"/>
        </codecs_info>
        <properties key="key1" value="val1"/>
      </media_profile>
    </ok>
  </response>
</out>
```

Нс_media_profile_declare

Команда для создания медиа-профилей в домене.

Аналог команды в CoCon:

domain/DOMAIN_NAME/media-profile/declare <name> <description> <priority>

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/media_profile_declare

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XML-схема:

media_profile.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <!-- MediaProfile -->
  <xs:complexType name="mediaProfileType">
    <xs:sequence>
      <xs:element name="codecs_info" type="mediaProfileCodecsType" minOccurs="0"
maxOccurs="unbounded"/>
      <xs:element name="properties" type="proplistType" minOccurs="0" maxOccurs="unbounded"
/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="codecs_priority" type="codecsPriorityType" use="required"/>
  </xs:complexType>

  <xs:simpleType name="codecsPriorityType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="user"/>
      <xs:enumeration value="system"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="proplistType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- MediaProfileCodecs -->
  <xs:complexType name="mediaProfileCodecsType">
    <xs:sequence>
      <xs:element name="codecs" type="mediaProfileCodecType" minOccurs="0"
maxOccurs="unbounded"/>
      <xs:element name="properties" type="proplistType" minOccurs="0" maxOccurs="unbounded"
/>
    </xs:sequence>
    <xs:attribute name="type" type="codecsType" use="required"/>
  </xs:complexType>

  <xs:simpleType name="codecsType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="audio"/>
      <xs:enumeration value="video"/>
      <xs:enumeration value="image"/>
      <xs:enumeration value="<other>"/>
    </xs:restriction>
  </xs:simpleType>

```

```
<!-- MediaProfilecodec -->
<xs:complexType name="mediaProfileCodecType">
  <xs:sequence>
    <xs:element name="attributes" type="proplistType" minOccurs="0" maxOccurs="unbounded"
  />
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/> <!-- or special codec
wildcard <other> -->
  <xs:attribute name="payload_type" type="xs:string" use="required"/> <!-- or special PT
wildcard * -->
  <xs:attribute name="enabled" type="xs:boolean" use="required"/>
</xs:complexType>

  <xs:element name="media_profile" type="mediaProfileType"/>

</xs:schema
```

XML-схема:**media_profile_declare.xsd**

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="media_profile.xsd"/>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="media_profile" type="mediaProfileType" minOccurs="1"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok" type="okType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>

  <xs:complexType name="okType"/>

  <xs:complexType name="errorType">
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Примеры:

http://192.168.1.21:9999/commands/media_profile_declare

Запрос:

```
<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request>
    <media_profile domain="biysk.local" name="mp1" description="add_video"
codecs_priority="system">
      <codecs_info type="video">
        <codecs name="H264" payload_type="*" enabled="true">
          <attributes key="key3" value="val3"/>
        </codecs>
        <properties key="offroad" value="true"/>
      </codecs_info>
      <properties key="key1" value="val1"/>
    </media_profile>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_media_profile_declare.xsd">
  <response>
    <ok/>
  </response>
</out>
```

[Hc_media_profile_info](#)

Команда для просмотра параметров медиа-профиля.

Аналог команды в CoCon:

domain/DOMAIN_NAME/media-profile/info <name>

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/media_profile_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="media_profile.xsd"/>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="profile_name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="media_profile" type="mediaProfileType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>

  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Примеры:

http://192.168.1.21:9999/commands/media_profile_info

Запрос:

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local" profile_name="mp_video"/>
</in>
```

ОТВЕТ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_profile_info.xsd">
  <response>
    <media_profile domain="biysk.local" name="mp_video" description="video"
  codecs_priority="system">
      <codecs_info type="<other>">
        <codecs name="<other>" payload_type="*" enabled="true"/>
        <properties key="offroad" value="true"/>
      </codecs_info>
      <codecs_info type="audio">
        <codecs name="G722" payload_type="9" enabled="true"/>
        <codecs name="PCMA" payload_type="8" enabled="true"/>
        <codecs name="PCMU" payload_type="0" enabled="true"/>
        <properties key="offroad" value="true"/>
      </codecs_info>
      <codecs_info type="video">
        <codecs name="H263-1998" payload_type="*" enabled="true"/>
        <codecs name="H264" payload_type="*" enabled="true"/>
        <properties key="offroad" value="true"/>
      </codecs_info>
    </media_profile>
  </response>
</out>
```

hc_media_profile_set

Команда для изменения общих параметров медиа-профиля.

Аналог команды в CoCon:

domain/DOMAIN_NAME/media-profile/set <name> <param> <value>

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/media_profile_set

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="media_profile.xsd"/>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="media_profile" type="mediaProfileType" minOccurs="1"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok" type="okType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>

  <!-- okType -->
  <xs:complexType name="okType"/>

  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

</xs:schema

```

Примеры:

http://192.168.1.21:9999/commands/media_profile_set

Запрос:

```
<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  <request>
    <media_profile domain="biysk.local" name="mp_video" description="for office"
codecs_priority="user">
      <codecs_info type="audio">
        <codecs name="G729" payload_type="18" enabled="true">
          <attributes key="offroad" value="true"/>
        </codecs>
        <properties key="key2" value="val2"/>
      </codecs_info>
      <properties key="key1" value="val1"/>
    </media_profile>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_media_profile_set.xsd">
  <response>
    <ok/>
  </response>
</out>
```

[Hc_media_profile_remove](#)

Команда для удаления медиа-профиля.

Аналог команды в CoCon:

domain/DOMAIN_NAME/media-profile/remove <name>

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/media_profile_remove

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;

- 404 – в случае ошибки.

XML-схема:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="profile_name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok" type="okType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>

  <!-- okType -->
  <xs:complexType name="okType"/>

  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

</xs:schema>
```

Примеры:

http://192.168.1.21:9999/commands/media_profile_remove

Запрос:

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local" profile_name="mp1"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_profile_remove.xsd">
  <response>
    <ok/>
  </response>
</out>
```

Запрос:

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local" profile_name="test"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_profile_remove.xsd">
  <response>
    <error description="not_exists"/>
  </response>
</out>
```

Hc_media_registrar - Команды управления регистратором для MSR медиа менеджера

- [Hc_media_registrar_clean](#)
- [Hc_media_registrar_info](#)

- [Hc_media_registrar_set](#)

[Hc_media_registrar_clean](#)

Команда для сброса в значение по умолчанию настроек подключения медиа-ресурсов.

Аналог команды в CoCon:

`/system/media/registrar/clean`

Шаблон URL запроса:

POST

http://host:port/commands/media_registrar_clean

Код ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of media registrar.
      Usage: http://server:9999/commands/media_registrar_clean.
      Implementation: hc_media_registrar_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="optionType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="nodeOptionType">
    <xs:attribute name="node" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="option" type="optionType" minOccurs="0" maxOccurs="1"/>
              <xs:element name="node_option" type="nodeOptionType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="success" type="xs:string"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```
</xs:schema>
```

Пример

http://192.168.1.21/commands/media_registrar_clean

Запрос:

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_media_registrar_clean.xsd">
  <request>
    <option name="registration_timeout"/>
    <node_option node="core1@ecss1" name="is_active"/>
    <node_option node="core1@ecss1" name="listen_ip"/>
    <node_option node="core1@ecss1" name="listen_port"/>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_registrar_clean.xsd">
  <success/>
</out>
```

[Hc_media_registrar_info](#)

Команда предназначена для просмотра настроек подключения медиа-ресурсов.

Аналог команды в CoCon:

/system/media/registrar/info

Шаблон URL-запроса:

POST

http://host:port/commands/media_registrar_info

Код ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of media registrar.
      Usage: http://server:9999/commands/media_registrar_info.
      Implementation: hc_media_registrar_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- commonOptionsType -->
  <xs:complexType name="commonOptionsType">
    <xs:attribute name="registration_timeout" type="xs:integer" use="required"/>
  </xs:complexType>

  <!-- nodeOptionsType -->
  <xs:complexType name="nodeOptionsType">
    <xs:attribute name="node" type="xs:string" use="required"/>
    <xs:attribute name="status" type="xs:string" use="required"/>
    <xs:attribute name="is_active" type="xs:boolean" use="required"/>
    <xs:attribute name="listen_ip" type="xs:string" use="required"/>
    <xs:attribute name="listen_port" type="xs:integer" use="required"/>
  </xs:complexType>

  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:element name="commonParameters" type="commonOptionsType" minOccurs="1"/>
      <xs:element name="privateParameters" type="nodeOptionsType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType/>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">

```

```

<xs:complexType>
  <xs:choice>
    <xs:element name="result" type="resultType"/>
    <xs:element name="error" type="errorType"/>
  </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema

```

Пример

http://192.168.1.21:9999/commands/media_registrar_info

Запрос:

```

<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_media_registrar_info.xsd">
  <request/>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_registrar_info.xsd">
  <result>
    <commonParameters registration_timeout="90"/>
    <privateParameters node="core1@ecss1" status="active" is_active="true"
listen_ip="192.168.2.21" listen_port="5000"/>
    <privateParameters node="core1@ecss2" status="active" is_active="true"
listen_ip="192.168.2.22" listen_port="5000"/>
  </result>
</out>

```

Hc_media_registrar_set

Команда предназначена для редактирования настроек подключения медиа-ресурсов.

Аналог команды в CoCon:

/system/media/registrar/set

Шаблон URL запроса:

POST

http://host:port/commands/media_registrar_set

Коды HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of media registrar.
      Usage: http://server:9999/commands/media_registrar_clean.
      Implementation: hc_media_registrar_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

<!-- resultType-->
<xs:complexType name="resultType">
  <xs:choice>
    <xs:element name="success" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="error" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
  </xs:choice>
</xs:complexType>

<xs:complexType name="optionType">
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="value" type="xs:string" use="required"/>
</xs:complexType>

<xs:complexType name="nodeOptionType">
  <xs:attribute name="node" type="xs:string" use="required"/>
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="value" type="xs:string" use="required"/>
</xs:complexType>

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="option" type="optionType" minOccurs="0" maxOccurs="1"/>
            <xs:element name="node_option" type="nodeOptionType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>

```

```

</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

http://192.168.1.21:9999/commands/media_registrar_set

Запрос:

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_media_registrar_set.xsd">
  <request>
    <option name="registration_timeout" value="60"/>
    <node_option node="core1@ecss1" name="is_active" value="true"/>
    <node_option node="core1@ecss1" name="listen_port" value="5001"/>
    <node_option node="core1@ecss1" name="listen_ip" value="0.0.0.0"/>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_registrar_set.xsd">
  <result/>
</out>

```

Ошибка:

```

<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_registrar_set.xsd">
  <error cmd="media_registrar_set" reason="error" body="{error,bad_node,core1@ecss3}"/>
</out>

```

Hc_media_resource - Команды управления медиа-ресурсами

- [Hc_media_resource_declare](#)
- [Hc_media_resource_info](#)
- [Hc_media_resource_purge](#)
- [Hc_media_resource_remove](#)
- [Hc_media_resource_set](#)

В данном разделе описаны команды по управлению подключенными медиа-серверами к ECSS-10.

[Hc_media_resource_declare](#)

Команда для декларации подключения медиа-сервера к ECSS-10.

Аналог команды в CoCon:

`/system/media/resource/declare`

Шаблон URL запроса:

http://host:port/commands/media_resource_declare

Коды ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Declares properties of media resource.
      Usage: http://server:9999/commands/media_resource_declare.
      Implementation: hc_media_registrar_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:choice maxOccurs="unbounded">
        <xs:element name="declared" type="mediaResourceType" minOccurs="0" maxOccurs="unbounded"/
>
        <xs:element name="error" type="mediaErrorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="mediaErrorType">
    <xs:attribute name="reason" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="mediaResourceType">
    <xs:attribute name="node" type="xs:string" use="required"/>
    <xs:attribute name="uri" type="xs:string" use="required"/>
    <xs:attribute name="zone" type="xs:string" use="required"/>
    <xs:attribute name="site" type="xs:string" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
  </xs:complexType>
  <xs:complexType name="declaredMediaResourceType">
    <xs:attribute name="property" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="errorMediaResourceType">
    <xs:attribute name="property" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>

```

```

        <xs:element name="media_resource" type="mediaResourceType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema

```

Пример**Запрос:**

```

<?xml version="1.0" encoding="UTF-8"?>
<in
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_media_resource_declare.xsd">
    <request>
        <media_resource site="local" zone="default" uri="bond1.476@msr_2" node="core1@ecss1"
        active="true"/>
    </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
    <out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="hc_media_resource_declare.xsd">
        <result/>
    </out>

```

Ошибка:


```
<?xml version="1.0"?>
  <out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_resource_declare.xsd">
    <result>
      <error reason="already_declared" value="bond1.2@msr_1"/>
    </result>
  </out>
```

[Hc_media_resource_info](#)

Команда для просмотра информации о подключении медиа-ресурса к ECSS-10.

Аналог команды в CoCon:

`/system/media/resource/info`

Шаблон URL запроса:

http://host:port/commands/media_resource_info

Коды ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of media resource.
      Usage: http://server:9999/commands/media_resource_info.
      Implementation: hc_media_resource_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

<!-- mediaResourceType -->
  <xs:complexType name="mediaResourceType">
    <xs:attribute name="node" type="xs:string" use="required"/>
    <xs:attribute name="sip_uri" type="xs:string" use="required"/>
    <xs:attribute name="sip_status" type="xs:string" use="required"/>
    <xs:attribute name="cc_status" type="xs:string" use="required"/>
    <xs:attribute name="cc_id" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="register_time" type="xs:string" use="required"/>
    <xs:attribute name="expired" type="xs:string" use="required"/>
    <xs:attribute name="iface" type="xs:string" use="required"/>
    <xs:attribute name="declare" type="xs:boolean" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
    <xs:attribute name="zone" type="xs:string" use="required"/>
    <xs:attribute name="site" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:element name="media_resource" type="mediaResourceType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
<!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType/>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>
```

Пример

http://192.168.1.21:9999/commands/media_resource_info

Запрос:

```
<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_media_resource_info.xsd">
  <request/>
</in>
```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_resource_info.xsd">
  <result>
    <media_resource node="core1@ecss1" sip_uri="bond1.2:SIP61@msr_1" sip_status="registered"
      cc_status="connected" cc_id="22abcd" address="" register_time="2019/11/13 15:57:30" expired="88"
      iface="bond1.2:SIP61(192.168.2.61)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss1" sip_uri="bond1.2:SIP62@msr_2" sip_status="registered"
      cc_status="connected" cc_id="c5b9d8" address="" register_time="2019/11/13 15:58:32"
      expired="150" iface="bond1.2:SIP62(192.168.2.62)" declare="false" active="false" zone="" site=""/>
  >
    <media_resource node="core1@ecss1" sip_uri="bond1.2@msr_1" sip_status="registered"
      cc_status="connected" cc_id="22abcd" address="192.168.2.21:5700" register_time="2019/11/13
      15:57:30" expired="88" iface="bond1.2(192.168.2.21)" declare="true" active="true" zone="default"
      site="local"/>
    <media_resource node="core1@ecss1" sip_uri="bond1.2@msr_2" sip_status="registered"
      cc_status="connected" cc_id="c5b9d8" address="192.168.2.22:5700" register_time="2019/11/13
      15:57:36" expired="94" iface="bond1.2(192.168.2.22)" declare="true" active="true" zone="default"
      site="local"/>
    <media_resource node="core1@ecss1" sip_uri="bond1.3:MYSQL@msr_1" sip_status="registered"
      cc_status="connected" cc_id="22abcd" address="" register_time="2019/11/13 15:57:30" expired="88"
      iface="bond1.3:MYSQL(192.168.1.71)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss1" sip_uri="bond1.3@msr_1" sip_status="registered"
      cc_status="connected" cc_id="22abcd" address="" register_time="2019/11/13 15:57:30" expired="88"
      iface="bond1.3(192.168.1.21)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss1" sip_uri="bond1.3@msr_2" sip_status="registered"
      cc_status="connected" cc_id="c5b9d8" address="" register_time="2019/11/13 15:57:36" expired="94"
      iface="bond1.3(192.168.1.22)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss1" sip_uri="bond1.476@msr_1" sip_status="registered"
      cc_status="connected" cc_id="22abcd" address="" register_time="2019/11/13 15:57:30" expired="88"
      iface="bond1.476(10.16.33.5)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss1" sip_uri="bond1.476@msr_2" sip_status="registered"
      cc_status="connected" cc_id="c5b9d8" address="" register_time="2019/11/13 15:57:36" expired="94"
      iface="bond1.476(10.16.33.6)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss1" sip_uri="lo@msr_1" sip_status="registered"
      cc_status="connected" cc_id="22abcd" address="" register_time="2019/11/13 15:57:30" expired="88"
      iface="lo(127.0.0.1)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss1" sip_uri="lo@msr_2" sip_status="registered"
      cc_status="connected" cc_id="c5b9d8" address="" register_time="2019/11/13 15:57:36" expired="94"
      iface="lo(127.0.0.1)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss2" sip_uri="bond1.2:SIP61@msr_1" sip_status="registered"
      cc_status="connected" cc_id="22abcd" address="" register_time="2019/11/13 15:57:30" expired="88"
      iface="bond1.2:SIP61(192.168.2.61)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss2" sip_uri="bond1.2:SIP62@msr_2" sip_status="registered"
      cc_status="connected" cc_id="c5b9d8" address="" register_time="2019/11/13 15:57:36" expired="94"
      iface="bond1.2:SIP62(192.168.2.62)" declare="false" active="false" zone="" site=""/>
  </result>
</out>

```

```

    <media_resource node="core1@ecss2" sip_uri="bond1.2@msr_1" sip_status="registered"
    cc_status="connected" cc_id="22abcd" address="192.168.2.21:5700" register_time="2019/11/13
    15:57:30" expired="88" iface="bond1.2(192.168.2.21)" declare="true" active="true" zone="default"
    site="local"/>
    <media_resource node="core1@ecss2" sip_uri="bond1.2@msr_2" sip_status="registered"
    cc_status="connected" cc_id="c5b9d8" address="192.168.2.22:5700" register_time="2019/11/13
    15:57:37" expired="95" iface="bond1.2(192.168.2.22)" declare="true" active="true" zone="default"
    site="local"/>
    <media_resource node="core1@ecss2" sip_uri="bond1.3:MYSQL@msr_1" sip_status="registered"
    cc_status="connected" cc_id="22abcd" address="" register_time="2019/11/13 15:57:30" expired="88"
    iface="bond1.3:MYSQL(192.168.1.71)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss2" sip_uri="bond1.3@msr_1" sip_status="registered"
    cc_status="connected" cc_id="22abcd" address="" register_time="2019/11/13 15:57:30" expired="88"
    iface="bond1.3(192.168.1.21)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss2" sip_uri="bond1.3@msr_2" sip_status="registered"
    cc_status="connected" cc_id="c5b9d8" address="" register_time="2019/11/13 15:57:37" expired="95"
    iface="bond1.3(192.168.1.22)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss2" sip_uri="bond1.476@msr_1" sip_status="registered"
    cc_status="connected" cc_id="22abcd" address="" register_time="2019/11/13 15:57:30" expired="88"
    iface="bond1.476(10.16.33.5)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss2" sip_uri="bond1.476@msr_2" sip_status="registered"
    cc_status="connected" cc_id="c5b9d8" address="" register_time="2019/11/13 15:57:37" expired="95"
    iface="bond1.476(10.16.33.6)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss2" sip_uri="lo@msr_1" sip_status="registered"
    cc_status="connected" cc_id="22abcd" address="" register_time="2019/11/13 15:57:30" expired="88"
    iface="lo(127.0.0.1)" declare="false" active="false" zone="" site=""/>
    <media_resource node="core1@ecss2" sip_uri="lo@msr_2" sip_status="registered"
    cc_status="connected" cc_id="c5b9d8" address="" register_time="2019/11/13 15:57:37" expired="95"
    iface="lo(127.0.0.1)" declare="false" active="false" zone="" site=""/>
  </result>
</out>

```

Нс_media_resource_purge

Команда для удаления старых, не активных подключений медиа-ресурсов к ECSS-10.

Аналог команды в CoCon:

/system/media/resource/purge

Шаблон URL запроса:

http://host:port/commands/media_resource_purge

Коды ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Purge properties of media resource.
      Usage: http://server:9999/commands/media_resource_purge.
      Implementation: hc_media_resource_purge.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

<!-- resultType -->
<xs:complexType name="resultType">
  <xs:sequence>
    <xs:choice maxOccurs="unbounded">
      <xs:element name="success" type="nodeResType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="error" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
    </xs:choice>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="nodeType">
  <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>

<xs:complexType name="sipUriType">
  <xs:attribute name="sip_uri" type="xs:string" use="required"/>
</xs:complexType>

<xs:complexType name="nodeResType">
  <xs:sequence>
    <xs:element name="deleted" type="sipUriType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="node" type="xs:string" use="required"/>
</xs:complexType>

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="node" type="nodeType" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

http://192.168.1.21/commands/media_resource_purge

Запрос:

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_media_resource_purge.xsd">
  <request>
    <node name="core1@ecss2"/>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_resource_purge.xsd">
  <result>
    <success node="core1@ecss2"/>
  </result>
</out>

```

Нс_media_resource_remove

Команда для удаления информации о настройках медиа-ресурса, добавленных ранее в ECSS-10.

Аналог команды в CoCon:

`/system/media/resource/remove`

Шаблон URL запроса:

`http://host:port/commands/media_resource_remove`

Коды ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Removes properties of media resource.
      Usage: http://server:9999/commands/media_resource_remove.
      Implementation: hc_media_resource_remove.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

<!-- resultType -->
<xs:complexType name="resultType">
  <xs:sequence>
    <xs:choice maxOccurs="unbounded">
      <xs:element name="deleted" type="sipUriType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:choice>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="resourceType">
  <xs:attribute name="node" type="xs:string" use="required"/>
  <xs:attribute name="uri" type="xs:string" use="required"/>
</xs:complexType>

<xs:complexType name="sipUriType">
  <xs:attribute name="sip_uri" type="xs:string" use="required"/>
</xs:complexType>

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="resource" type="resourceType" minOccurs="0" maxOccurs="unbounded"
/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema

```

Примеры

http://192.168.1.21:9999/commands/media_resource_remove

Запрос:

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_media_resource_remove.xsd">
  <request>
    <resource node="core1@ecss2" uri=" bond1.2@msr_2 "/>
    <resource node="core1@ecss2" uri="bond1.476@msr_2"/>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_resource_remove.xsd">
  <result>
    <deleted sip_uri="bond1.476@msr_2"/>
    <deleted sip_uri=" bond1.2@msr_2 "/>
  </result>
</out>

```

hc_media_resource_set

Команда для изменения настроек медиа-ресурсов.

Аналог команды в CoCon:

/system/media/resource/set

Шаблон URL запроса:

http://host:port/commands/media_resource_set

Коды ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set properties of media resource.
      Usage: http://server:9999/commands/media_resource_set.
      Implementation: hc_media_resource_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

<!-- resultType -->
  <xs:complexType name="resultType">
    <xs:choice>
      <xs:element name="media_resource" type="mediaResourceResultType" minOccurs="0"
maxOccurs="unbounded"/>
      <xs:element name="error" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
    </xs:choice>
  </xs:complexType>

  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="mediaResourceType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="node" type="xs:string" use="required"/>
    <xs:attribute name="uri" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="mediaResourceResultType">
    <xs:choice>
      <xs:sequence>
        <xs:element name="success" type="successPropertyResultType" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="error" type="errorPropertyResultType" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:choice>
    <xs:attribute name="node" type="xs:string" use="required"/>
    <xs:attribute name="uri" type="xs:string" use="required"/>
  </xs:complexType>

```

```

<xs:complexType name="successPropertyResultType">
  <xs:attribute name="property" type="xs:string" use="required"/>
</xs:complexType>

<xs:complexType name="errorPropertyResultType">
  <xs:attribute name="property" type="xs:string" use="required"/>
  <xs:attribute name="reason" type="xs:string" use="required"/>
</xs:complexType>

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="media_resource" type="mediaResourceType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Примеры

http://192.168.1.21:9999/commands/media_resource_set

Запрос:

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_media_resource_set.xsd">
  <request>
    <media_resource node="core1@ecss2" uri="bond1.2@msr_2">
      <property value="true" key="active"/>
      <property value="local" key="site"/>
      <property value="default" key="zone"/>
    </media_resource>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_resource_set.xsd">
  <result>
    <media_resource node="core1@ecss2" uri="bond1.2@msr_2">
      <success property="active"/>
      <success property="site"/>
      <success property="zone"/>
    </media_resource>
  </result>
</out>
```

Hc_media_site - Команды управления географическими зонами обслуживания медиа-сервера

- [Hc_media_site_declare](#)
- [Hc_media_site_distance](#)
- [Hc_media_site_list](#)
- [Hc_media_site_matrix](#)
- [Hc_media_site_remove](#)

Hc_media_site_declare

Команда предназначена для создания новой географической зоны (нового сайта), которую будет обслуживать медиа-сервер.

Аналог команды в CoCon:

system/media/site/declare

Шаблон URL запроса:

POST

http://host:port/commands/media_site_declare

Коды ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Declares media site.
      Usage: http://server:9999/commands/media_site_declare.
      Implementation: hc_media_site_declare.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

<!-- mediaSiteType -->
  <xs:complexType name="mediaSiteType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="mediaSiteResultType">
    <xs:attribute name="site" type="xs:string" use="required"/>
  </xs:complexType>

<!-- resultType-->
  <xs:complexType name="resultType">
    <xs:choice maxOccurs="unbounded">
      <xs:sequence>
        <xs:element name="created" type="mediaSiteResultType" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="error" type="errorMediaSiteResultType" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:choice>
  </xs:complexType>

  <xs:complexType name="errorMediaSiteResultType">
    <xs:attribute name="reason" type="xs:string" use="required"/>
    <xs:attribute name="property" type="xs:string" use="required"/>
  </xs:complexType>

<!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>

```



```

        <xs:element name="site" type="mediaSiteType" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:all>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

http://192.168.1.21:9999/commands/media_site_declare

Запрос:

```

<?xml version="1.0"?>
  <in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_media_site_declare.xsd">
    <request>
      <site name="core-site1"/>
      <site name="core-site2"/>
      <site name="core-site3"/>
      <site name="core-site4"/>
      <site name="core-site5"/>
      <site name="core-site6"/>
    </request>
  </in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_site_declare.xsd">
  <result>
    <error reason="already_declared" property="core-site1"/>
    <error reason="already_declared" property="core-site2"/>
    <error reason="already_declared" property="core-site3"/>
    <created site="core-site4"/>
    <created site="core-site5"/>
    <created site="core-site6"/>
  </result>
</out>
```

[Hc_media_site_distance](#)

Команда предназначена для установки "расстояния" между сайтами.

Аналог команды в CoCon:

system/media/site/distance

Шаблон URL запроса:

POST

http://host:port/commands/media_site_distance

Коды ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Declares media site.
      Usage: http://server:9999/commands/media_site_declare.
      Implementation: hc_media_site_declare.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

<!-- mediaSiteType -->
  <xs:complexType name="mediaSiteType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="mediaSiteResultType">
    <xs:attribute name="site" type="xs:string" use="required"/>
  </xs:complexType>

<!-- resultType-->
  <xs:complexType name="resultType">
    <xs:choice maxOccurs="unbounded">
      <xs:sequence>
        <xs:element name="created" type="mediaSiteResultType" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="error" type="errorMediaSiteResultType" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:choice>
  </xs:complexType>

  <xs:complexType name="errorMediaSiteResultType">
    <xs:attribute name="reason" type="xs:string" use="required"/>
    <xs:attribute name="property" type="xs:string" use="required"/>
  </xs:complexType>

<!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>

```

```

        <xs:element name="site" type="mediaSiteType" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:all>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

http://192.168.1.21:9999/commands/media_site_distance

Запрос:

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_site_distance.xsd">
  <request>
    <arc site_l="core-site1" site_r="core-site1" distance="1"/>
    <arc site_l="core-site1" site_r="core-site2" distance="3"/>
    <arc site_l="core-site1" site_r="core-site3" distance="4"/>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_site_distance.xsd">
  <result>
    <success site_l="core-site1" site_r="core-site1" distance="1"/>
    <success site_l="core-site1" site_r="core-site2" distance="3"/>
    <success site_l="core-site1" site_r="core-site3" distance="4"/>
  </result>
</out>

```

Нс_media_site_list

Команда возвращает список географических зон (сайтов), обслуживаемых медиа-серверами.

Аналог команды в CoCon:

system/media/site/list

Шаблон URL запроса:

POST

http://host:port/commands/media_site_list

Коды ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Shows media site list.
      Usage: http://server:9999/commands/media_site_list.
      Implementation: hc_media_site_list.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

<!-- mediaSiteType -->
  <xs:complexType name="mediaSiteType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

<!-- resultType-->
  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:element name="site" type="mediaSiteType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
<!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType/>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
<!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

http://192.168.1.21:9999/commands/media_site_list

Запрос:

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_media_site_list.xsd">
  <request/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_site_list.xsd">
  <result>
    <site name="core-site1"/>
    <site name="core-site2"/>
    <site name="core-site3"/>
    <site name="core-site4"/>
    <site name="core-site5"/>
    <site name="core-site6"/>
    <site name="local"/>
  </result>
</out>
```

[Hc_media_site_matrix](#)

Команда позволяет просмотреть матрицу географических зон, которые обслуживает медиа-сервер.

Аналог команды в CoCon:

system/media/site/matrix

Шаблон URL запроса:

POST

http://host:port/commands/media_site_matrix

Коды ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Matrix of distances between sites.
      Usage: http://server:9999/commands/media_site_matrix.
      Implementation: hc_media_site_matrix.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

<!-- mediaSiteType -->
  <xs:complexType name="siteMatrixType">
    <xs:attribute name="site_l" type="xs:string" use="required"/>
    <xs:attribute name="site_r" type="xs:string" use="required"/>
    <xs:attribute name="distance" type="xs:string" use="required"/>
  </xs:complexType>

<!-- resultType-->
  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:element name="arc" type="siteMatrixType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

<!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType/>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

<!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```


Пример

http://192.168.1.21:9999/commands/media_site_matrix

Запрос:

```
<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_site_matrix.xsd">
  <request/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_site_matrix.xsd">
  <result>
    <arc site_l="core-site1" site_r="core-site1" distance="1"/>
    <arc site_l="core-site1" site_r="core-site2" distance="3"/>
    <arc site_l="core-site1" site_r="core-site3" distance="4"/>
    <arc site_l="core-site2" site_r="core-site2" distance="0"/>
    <arc site_l="core-site2" site_r="core-site1" distance="3"/>
    <arc site_l="core-site3" site_r="core-site3" distance="0"/>
    <arc site_l="core-site3" site_r="core-site1" distance="4"/>
    <arc site_l="core-site4" site_r="core-site4" distance="0"/>
    <arc site_l="core-site5" site_r="core-site5" distance="0"/>
    <arc site_l="core-site6" site_r="core-site6" distance="0"/>
    <arc site_l="local" site_r="local" distance="0"/>
  </result>
</out>
```

[Hc_media_site_remove](#)

Команда позволяет удалить сайт, обслуживаемый медиа-сервером, и всю информацию о нем.

Аналог команды в CoCon:

system/media/site/remove

Шаблон URL запроса:

POST

http://host:port/commands/media_site_remove

Коды ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Remove site.
      Usage: http://server:9999/commands/media_site_remove.
      Implementation: hc_media_site_remove.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

<!-- mediaSiteType -->
  <xs:complexType name="mediaSiteType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="mediaSiteResultType">
    <xs:attribute name="site" type="xs:string" use="required"/>
  </xs:complexType>

<!-- resultType-->
  <xs:complexType name="resultType">
    <xs:choice maxOccurs="unbounded">
      <xs:sequence>
        <xs:element name="deleted" type="mediaSiteResultType" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:choice>
  </xs:complexType>

<!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="site" type="mediaSiteType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

http://192.168.1.21:9999/commands/media_site_remove

Запрос:

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_site_remove.xsd">
  <request>
    <site name="core-site4"/>
    <site name="core-site5"/>
    <site name="core-site6"/>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_site_remove.xsd">
  <result>
    <deleted site="core-site6"/>
    <deleted site="core-site5"/>
    <deleted site="core-site4"/>
  </result>
</out>

```

Hc_media_zone - Команды управления идентификатором сети

- [Hc_media_zone_declare](#)
- [Hc_media_zone_list](#)
- [Hc_media_zone_remove](#)

Нс_media_zone_declare

Команда предназначена для создания нового идентификатора сети для медиа-сервера.

Аналог команды в CoCon:

system/media/zone/declare

Шаблон URL запроса:

POST

http://host:port/commands/media_zone_declare

Коды ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Declare zone.
      Usage: http://server:9999/commands/media_zone_declare.
      Implementation: hc_media_zone_declare.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

<!-- mediaZoneType -->
  <xs:complexType name="mediaZoneType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="mediaZoneResultType">
    <xs:attribute name="zone" type="xs:string" use="required"/>
  </xs:complexType>

<!-- resultType-->
  <xs:complexType name="resultType">
    <xs:choice maxOccurs="unbounded">
      <xs:sequence>
        <xs:element name="created" type="mediaZoneResultType" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="error" type="errorMediaZoneResultType" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:choice>
  </xs:complexType>

  <xs:complexType name="errorMediaZoneResultType">
    <xs:attribute name="reason" type="xs:string" use="required"/>
    <xs:attribute name="property" type="xs:string" use="required"/>
  </xs:complexType>

<!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>

```

```

        <xs:element name="zone" type="mediaZoneType" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:all>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

http://192.168.1.21:9999/commands/media_zone_declare

Запрос:

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_zone_declare.xsd">
  <request>
    <zone name="core-zone1"/>
    <zone name="core-zone2"/>
    <zone name="core-zone3"/>
    <zone name="core-zone4"/>
    <zone name="core-zone5"/>
    <zone name="core-zone6"/>
  </request>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_zone_declare.xsd">
  <result>
    <created zone="core-zone1"/>
    <created zone="core-zone2"/>
    <created zone="core-zone3"/>
    <created zone="core-zone4"/>
    <created zone="core-zone5"/>
    <created zone="core-zone6"/>
  </result>
</out>
```

[Hc_media_zone_list](#)

Команда возвращает список идентификаторов сети для медиа-серверов.

Аналог команды в CoCon:

system/media/zone/list

Шаблон URL запроса:

POST

http://host:port/commands/media_zone_list

Коды ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      List of zones.
      Usage: http://server:9999/commands/media_zone_list.
      Implementation: hc_media_zone_list.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- mediaZoneType -->
  <xs:complexType name="mediaZoneType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:element name="zone" type="mediaZoneType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType/>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

http://192.168.1.21:9999/commands/media_zone_list

Запрос:

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_media_zone_list.xsd">
  <request/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_zone_list.xsd">
  <result>
    <zone name="core-zone1"/>
    <zone name="core-zone2"/>
    <zone name="core-zone3"/>
    <zone name="core-zone4"/>
    <zone name="core-zone5"/>
    <zone name="core-zone6"/>
    <zone name="default"/>
  </result>
</out>
```

hc_media_zone_remove

Команда предназначена для удаление идентификатора сети для медиа-сервера.

Аналог команды в CoCon:

system/media/zone/remove

Шаблон URL запроса:

POST

http://host:port/commands/media_zone_remove

Коды ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Remove zone.
      Usage: http://server:9999/commands/media_zone_remove.
      Implementation: hc_media_zone_remove.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

<!-- mediaZoneType -->
  <xs:complexType name="mediaZoneType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="mediaZoneResultType">
    <xs:attribute name="zone" type="xs:string" use="required"/>
  </xs:complexType>

<!-- resultType-->
  <xs:complexType name="resultType">
    <xs:choice maxOccurs="unbounded">
      <xs:sequence>
        <xs:element name="deleted" type="mediaZoneResultType" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:choice>
  </xs:complexType>

<!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="zone" type="mediaZoneType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

http://192.168.1.21:9999/commands/media_zone_remove

Запрос:

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_zone_remove.xsd">
  <request>
    <zone name="core-zone1"/>
    <zone name="core-zone2"/>
    <zone name="core-zone6"/>
    <zone name="core-zone7"/>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_media_zone_remove.xsd">
  <result>
    <deleted zone="core-zone6"/>
    <deleted zone="core-zone2"/>
    <deleted zone="core-zone1"/>
    <error cmd="hc_media_zone_remove" reason="zone_not_found" entity="core-zone7"/>
  </result>
</out>

```

Нс_mediator_http_terminal - Управление подсистемой HTTP Terminal

- [Общая схема](#)

- [Hc_mediator_http_terminal_info](#)
- [Hc_mediator_http_terminal_set](#)
- [Hc_mediator_http_terminal_clean](#)

Общая схема

hc_mediator_common.xsd:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- Options for TTS node -->
  <xs:complexType name="httpTerminalOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="http_server_name" type="stringType" use="optional" />
        <xs:attribute name="http_server_port" type="inetPortType" use="optional" />
        <xs:attribute name="http_server_listen_iface" type="ipAddressType" use="optional" />
        <xs:attribute name="request_time" type="positiveIntegerType" use="optional" />
        <xs:attribute name="session_lifetime" type="positiveIntegerType" use="optional" />
        <xs:attribute name="ssl_certfile" type="stringType" use="optional" />
        <xs:attribute name="ssl_keyfile" type="stringType" use="optional" />
        <xs:attribute name="ssl_password" type="stringType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              In case of strong security restrictions password may be shown as *****
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>
```

[Hc_mediator_http_terminal_info](#)

Команда используется для получения свойств подсистемы HTTP Terminal кластера Mediator.

Аналог команды в CoCon:

cluster/mediator/<CLUSTER>/properties/cocon_http_terminal/info

Шаблон HTTP-запроса:

http://host:port/commands/mediator_http_terminal_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="mediator_http_terminal_info">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get options of TTS node.
      Usage: http://server:9999/commands/mediator\_http\_terminal\_info
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <xs:include schemaLocation="hc_mediator_common.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="nc_options" type="ncInfoInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="nc_options" type="ncInfoOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример 1:

Запрос:

http://192.168.1.21:9999/commands/mediator_http_terminal_info

```
<?xml version="1.0" encoding="UTF-8"?><in>
  <request>
    <nc_options cluster="md1"/>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_mediator_http_terminal_info.xsd">
  <result>
    <nc_options>
      <cluster name="md1">
        <options xs:type="httpTerminalOptionsType" ssl_password="*****"
          ssl_keyfile="undefined" ssl_certfile="undefined" session_lifetime="3600000" request_time="60000"
          http_server_port="9999" http_server_name="http_terminal" http_server_listen_iface="0.0.0.0"/>
      </cluster>
    </nc_options>
  </result>
</out>
```

Пример 2:

Запрос:

http://192.168.1.21:9999/commands/mediator_http_terminal_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request>
    <nc_options cluster="md1" node="md1@ecss1"/>
  </request>
</in>
```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_mediator_http_terminal_info.xsd">
  <result>
    <nc_options>
      <cluster name="md1">
        <options xs:type="httpTerminalOptionsType"/>
      </cluster>
      <node name="md1@ecss1">
        <options xs:type="httpTerminalOptionsType" ssl_password="*****"
        ssl_keyfile="undefined" ssl_certfile="undefined" session_lifetime="3600000" request_time="60000"
        http_server_port="9999" http_server_name="http_terminal" http_server_listen_iface="0.0.0.0"/>
      </node>
    </nc_options>
  </result>
</out>

```

hc_mediator_http_terminal_set

Команда установки свойства подсистемы HTTP Terminal кластера Mediator.

Аналог команды в CoCon:

cluster/mediator/<CLUSTER>/properties/cocon_http_terminal/set

Шаблон HTTP-запроса:

http://host:port/commands/mediator_http_terminal_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="mediator_http_terminal_set">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get options of TTS node.
      Usage: http://server:9999/commands/mediator_info_terminal_set
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <xs:include schemaLocation="hc_mediator_common.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="nc_options" type="ncSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="nc_options" type="ncSetOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

XSD-схема hc_mediator_common.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- Options for TTS node -->
  <xs:complexType name="httpTerminalOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="http_server_name" type="stringType" use="optional" />
        <xs:attribute name="http_server_port" type="inetPortType" use="optional" />
        <xs:attribute name="http_server_listen_iface" type="ipAddressType" use="optional" />
        <xs:attribute name="request_time" type="positiveIntegerType" use="optional" />
        <xs:attribute name="session_lifetime" type="positiveIntegerType" use="optional" />
        <xs:attribute name="ssl_certfile" type="stringType" use="optional" />
        <xs:attribute name="ssl_keyfile" type="stringType" use="optional" />
        <xs:attribute name="ssl_password" type="stringType" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/mediator_http_terminal_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_mediator_http_terminal_set.xsd">
  <request>
    <nc_options>
      <cluster name="md1">
        <options xs:type="httpTerminalOptionsType" session_lifetime="1800000"/>
      </cluster>
      <node name="md1@ecss1">
        <options xs:type="httpTerminalOptionsType" session_lifetime="1800000"/>
      </node>
    </nc_options>
  </request>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_mediator_http_terminal_set.xsd">
  <result>
    <nc_options>
      <cluster name="md1">
        <success option="session_lifetime"/>
      </cluster>
      <node name="md1@ecss1">
        <success option="session_lifetime"/>
      </node>
    </nc_options>
  </result>
</out>
```

Или 404 в случае, если после установки свойств http-terminal стал недоступен.

[Hc_mediator_http_terminal_clean](#)

Команда очистки свойств подсистемы HTTP Terminal кластера Mediator.

Аналог команды в CoCon:

cluster/mediator/<CLUSTER>/properties/cocon_http_terminal/clean

Шаблон HTTP-запроса:

http://host:port/commands/mediator_http_terminal_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="mediator_http_terminal_clean">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get options of TTS node.
      Usage: http://server:9999/commands/mediator_http_terminal_clean
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="nc_options" type="ncCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="nc_options" type="ncCleanOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/mediator_http_terminal_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_tts_general_options_clean.xsd">
  <request>
    <nc_options>
      <cluster name="md1">
        <option name="http_server_name"/>
      </cluster>
      <node name="md1@ecss1">
        <option name="http_server_name"/>
      </node>
    </nc_options>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_mediator_http_terminal_clean.xsd">
  <response>
    <nc_options>
      <cluster name="md1">
        <success option="http_server_name"/>
      </cluster>
      <node name="md1@ecss1">
        <success option="http_server_name"/>
      </node>
    </nc_options>
  </response>
</out>

```

Запрос:

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_mediator_http_terminal_clean.xsd">
  <request>
    <nc_options>
      <cluster name="md1">
      </cluster>
    </nc_options>
  </request>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_mediator_http_terminal_clean.xsd">
  <result>
    <nc_options>
      <cluster name="md1"/>
    </nc_options>
  </result>
</out>
```

Нс_megaco_gateways - Управление настройками Megaco-шлюзов

- [Нс_megaco_gateways_add](#)
- [Нс_megaco_gateways_list](#)
- [Нс_megaco_gateways_remove](#)

Нс_megaco_gateways_add

Команда для добавления шлюза Megaco-адаптера.

Аналог команды в CoSnp:

domain/**<DOMAIN>**/**megaco/gateways/add**

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/megaco_gateways_add

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="gateway" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="gateway" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="megaco">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="gateway" type="gatewayType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="context" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok" type="okType" minOccurs="0"
maxOccurs="unbounded"/>
              <xs:element name="error" type="errorType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/megaco_gateways_add

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="megaco_gateways_add.xsd">
  <megaco domain="p.city" context="ctx_local">
    <gateway name="gw1"/>
    <gateway name="gw2"/>
    <gateway name="gw3"/>
  </megaco>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="megaco_gateways_add.xsd">
  <response>
    <ok gateway="gw2"/>
    <ok gateway="gw1"/>
    <error gateway="gw3" reason="not_found"/>
  </response>
</out>
```

Нс_megaco_gateways_list

Команда используется для получения списка шлюзов для Megaco-адаптера.

Аналог команды в CoCon:

domain/<DOMAIN>/megaco/gateways/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/megaco_gateways_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="megaco">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="gateway" type="gatewayType" minOccurs="0" maxOccurs="unbounded"
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/megaco_gateways_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="megaco_gateways_list.xsd">
  <megaco domain="p.city"/>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="megaco_gateways_list.xsd">
  <gateway name="gw1"/>
  <gateway name="gw2"/>
  <gateway name="gw3"/>
</out>

```

Нс_megaco_gateways_remove

Команда для удаления шлюза Megaco-адаптера.

Аналог команды в CoCon:

domain/<DOMAIN>/megaco/gateways/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/megaco_gateways_remove

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="gatewayType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="megaco">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="gateway" type="gatewayType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="force" type="xs:string" use="optional" default="false
"/>
            <xs:attribute name="delete_ifaces" type="xs:string" use="optional"
default="false"/>
            <xs:attribute name="delete_aliases" type="xs:string" use="optional"
default="false"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.23.34:9999/commands/megaco_gateways_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="megaco_gateways_remove.xsd">
  <megaco domain="p.city" force="true" delete_ifaces="true" delete_aliases="true">
    <gateway name="gw1"/>
    <gateway name="gw2"/>
    <gateway name="gw3"/>
  </megaco>
</in>
```

Ответ: 204

Нс_megaco - Управление настройками РА Megaco

- [Нс_megaco_info](#)
- [Нс_megaco_set](#)
- [Нс_megaco_clean](#)

Нс_megaco_info

Команда просмотра свойств Megaco в указанном домене.

Аналог команды в CoCon:

domain/**<DOMAIN>**/megaco/config/info

Шаблон URL-запроса:

http://server:9999/commands/megaco_info

Коды ответа:

- 200 - успех;
- 404 - ошибка.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="megaco_info">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get properties of MEGACO for the specified domain.
      Usage: http://server:9999/commands/megaco_info

      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

<!-- Types -->

<!-- Encodings -->
<xs:simpleType name="encodingType">
  <xs:restriction base="unionType">
    <xs:enumeration value="text" />
    <xs:enumeration value="pretty_text" />
    <xs:enumeration value="compact_text" />
    <xs:enumeration value="binary" />
  </xs:restriction>
</xs:simpleType>
<!-- Encodings -->

<!-- FaxDetect -->
<xs:simpleType name="FaxDetect">
  <xs:restriction base="unionType">
    <xs:enumeration value="callee" />
    <xs:enumeration value="caller" />
    <xs:enumeration value="both" />
    <xs:enumeration value="none" />
  </xs:restriction>
</xs:simpleType>
<!-- FaxDetect -->

<!-- a string like: nodename1@host,192.168.23.1;nodename2@host,192.168.23.2 -->
<xs:simpleType name="NodeInterfaces">
  <xs:restriction base="xs:string"/>
</xs:simpleType>

```

```

<!-- Options for MEGACO node -->
<xs:complexType name="megacoNodeOptionsType">
  <xs:complexContent>
    <xs:extension base="baseOptionsType">
      <xs:attribute name="transport" type="inetProtocolType" use="optional" />
      <xs:attribute name="listen_port" type="inetPortType" use="optional" />
      <xs:attribute name="bind_to" type="NodeInterfaces" use="optional" />
      <xs:attribute name="encoding" type="encodingType" use="optional" />
      <xs:attribute name="pkg_adid_dt" type="xs:integer" use="optional" />
      <xs:attribute name="fax_detect" type="FaxDetect" use="optional" />
      <xs:attribute name="send_release_ack" type="xs:boolean" use="optional" />
      <xs:attribute name="use_rtcp" type="xs:boolean" use="optional" />
      <xs:attribute name="enabled" type="xs:boolean" use="optional" />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<!-- Options for MEGACO node -->

<!-- Types -->

<!-- Root -->

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all minOccurs="1" maxOccurs="1">
      <xs:element name="request" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:all minOccurs="1" maxOccurs="1">
            <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1"
maxOccurs="1" />
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- in -->

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" minOccurs="1" maxOccurs="1">
        <xs:complexType>

```

```

        <xs:all minOccurs="1" maxOccurs="1">
            <xs:element name="domain_options" type="domainInfoOutType"
minOccurs="1"
                                maxOccurs="1" />
        </xs:all>
    </xs:complexType>
</xs:element>
    <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
</xs:choice>
</xs:complexType>
</xs:element>
<!-- out -->

<!-- Root -->

</xs:schema>

```

Пример:

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="megaco_info.xsd">
    <request>
        <domain_options domain="domain2" />
    </request>
</in>

```

Ответ:

```

<?xml version="1.0" encoding="UTF-8"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="megaco_info.xsd">
    <result>
        <domain_options>
            <system>
                <options xs:type="megacoNodeOptionsType"/>
            </system>
            <domain name="domain2">
                <options xs:type="megacoNodeOptionsType" use_rtcp="true" transport="tcp"
send_release_ack="true" pkg_adid_dt="100" listen_port="3456" fax_detect="both" encoding="text"
enabled="true" bind_to="node1@host1,192.168.23.1;node2@host2,192.168.23.2"/>
            </domain>
        </domain_options>
    </result>
</out>

```

Нс_megaco_set

Изменение свойств Megaco в указанном домене

Аналог команды в CoCon:

domain/<DOMAIN>/megaco/config/set

Шаблон URL-запроса:

http://server:9999/commands/megaco_set

Коды ответа:

- 200 - успех;
- 404 - ошибка.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="megaco_set">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set properties of MEGACO for the specified domain.
      Usage: http://server:9999/commands/megaco_set

      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

<!-- Types -->

<!-- Encodings -->
<xs:simpleType name="encodingType">
  <xs:restriction base="unionType">
    <xs:enumeration value="text" />
    <xs:enumeration value="pretty_text" />
    <xs:enumeration value="compact_text" />
    <xs:enumeration value="binary" />
  </xs:restriction>
</xs:simpleType>
<!-- Encodings -->

<!-- FaxDetect -->
<xs:simpleType name="FaxDetect">
  <xs:restriction base="unionType">
    <xs:enumeration value="callee" />
    <xs:enumeration value="caller" />
    <xs:enumeration value="both" />
    <xs:enumeration value="none" />
  </xs:restriction>
</xs:simpleType>
<!-- FaxDetect -->

<!-- a string like: nodename1@host,192.168.23.1;nodename2@host,192.168.23.2 -->
<xs:simpleType name="NodeInterfaces">
  <xs:restriction base="xs:string"/>
</xs:simpleType>

```



```

<!-- Options for MEGACO node -->
<xs:complexType name="megacoNodeOptionsType">
  <xs:complexContent>
    <xs:extension base="baseOptionsType">
      <xs:attribute name="transport" type="inetProtocolType" use="optional" />
      <xs:attribute name="listen_port" type="inetPortType" use="optional" />
      <xs:attribute name="bind_to" type="NodeInterfaces" use="optional" />
      <xs:attribute name="encoding" type="encodingType" use="optional" />
      <xs:attribute name="pkg_adid_dt" type="xs:integer" use="optional" />
      <xs:attribute name="fax_detect" type="FaxDetect" use="optional" />
      <xs:attribute name="send_release_ack" type="xs:boolean" use="optional" />
      <xs:attribute name="use_rtcp" type="xs:boolean" use="optional" />
      <xs:attribute name="enabled" type="xs:boolean" use="optional" />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<!-- Options for MEGACO node -->

<!-- Types -->

<!-- Root -->

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:complexContent>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:complexContent>
              <xs:all minOccurs="1" maxOccurs="1">
                <xs:element name="domain_options" type="domainSetInType"
minOccurs="1"
maxOccurs="1" />
              </xs:all>
            </xs:complexContent>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
<!-- in -->

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" minOccurs="1" maxOccurs="1">
        <xs:complexType>

```

```

        <xs:all minOccurs="1" maxOccurs="1">
            <xs:element name="domain_options" type="domainSetOutType"
minOccurs="1"
                                maxOccurs="1"/>
        </xs:all>
    </xs:complexType>
</xs:element>
    <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
</xs:choice>
</xs:complexType>
</xs:element>
<!-- out -->

<!-- Root -->

</xs:schema>

```

Пример:**Запрос:**

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="megaco_set.xsd">
    <request>
        <domain_options>
            <system>
                <options/>
            </system>
            <domain name="domain2">
                <options xs:type="megacoNodeOptionsType" use_rtcp="true" transport="tcp"
send_release_ack="true" pkg_adid_dt="100" listen_port="3456" fax_detect="both" encoding="text"
enabled="true" bind_to="node1@host1,192.168.23.1;node2@host2,192.168.23.2"/>
            </domain>
        </domain_options>
    </request>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="megaco_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="domain2">
        <success option="bind_to"/>
        <success option="enabled"/>
        <success option="encoding"/>
        <success option="fax_detect"/>
        <success option="listen_port"/>
        <success option="pkg_adid_dt"/>
        <success option="send_release_ack"/>
        <success option="transport"/>
        <success option="use_rtcp"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Нс_megaco_clean

Команда сброса свойств Мегасо в указанном домене.

Аналог команды в CoSoc:

domain/<DOMAIN>/megaco/config/clean

Шаблон URL-ответа:

http://server:9999/commands/megaco_clean

Коды ответа:

- 200 - успех;
- 404 - ошибка.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="megaco_clean">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean property of MEGACO for the specified domain.
      Usage: http://server:9999/commands/megaco_clean

      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="domain_options" type="domainCleanInType"
minOccurs="1"
                                maxOccurs="1" />
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="domain_options" type="domainCleanOutType"
minOccurs="1"
                                maxOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

        </xs:choice>
    </xs:complexType>
</xs:element>
<!-- out -->

</xs:schema>

```

Пример:**Запрос:**

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="megaco_clean.xsd">
  <request>
    <domain_options>
      <system/>
      <domain name="domain2">
        <option name="bind_to"/>
        <option name="enabled"/>
        <option name="encoding"/>
        <option name="fax_detect"/>
        <option name="listen_port"/>
        <option name="pkg_adid_dt"/>
        <option name="send_release_ack"/>
        <option name="transport"/>
        <option name="use_rtcp"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="megaco_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="domain2">
        <success option="bind_to"/>
        <success option="enabled"/>
        <success option="encoding"/>
        <success option="fax_detect"/>
        <success option="listen_port"/>
        <success option="pkg_adid_dt"/>
        <success option="send_release_ack"/>
        <success option="transport"/>
        <success option="use_rtcp"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Hc_modifiers_context - Управление контекстами модификации

- [Hc_modifiers_context_delete](#)
- [Hc_modifiers_context_import](#)
- [Hc_modifiers_context_list](#)
- [Hc_modifiers_context_show](#)

Hc_modifiers_context_delete

Команда предназначена для удаления контекста модификации номеров по имени из определенного домена.

Аналог команды в CoCon:

cluster/storage/<STORAGE>/modifiers/delete

domain/<DOMAIN>/modifiers/delete

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/modifiers_context_delete

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="context" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="error" type="errorType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

Удаление существующего контекста

Запрос:

http://192.168.1.21:9999/commands/modifiers_context_delete

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_modifiers_context_delete.xsd">
  <request domain="biysk.local" context="mod_1"/>
</in>

```

Ответ:

204

Пример 2:

Удаление несуществующего конекста

Запрос:

http://192.168.1.21:9999/commands/modifiers_context_delete

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_modifiers_context_delete.xsd">
  <request domain="biysk.local" context="mod2"/>
</in>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_modifiers_context_delete.xsd">
  <error cmd="modifiers_context_delete" reason="not_exists" body="not_found" entity="mod2"/>
</out>
```

Ответ:

404

hc_modifiers_context_import

Команда предназначена для импорта контекста модификации номеров в определенный домен. Если контекст в указанном именем уже существует - контекст будет обновлен.

Аналог команды в CoCon:

cluster/storage/<STORAGE>/modifiers/import

domain/<DOMAIN>/modifiers/import

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/modifiers_context_import

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />
  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- progressType -->
  <xs:complexType name="progressType">
    <xs:sequence>
      <xs:element name="ref" minOccurs="1" maxOccurs="1" type="valueType" />
      <xs:element name="text" minOccurs="1" maxOccurs="1" type="valueType" />
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="context" type="xs:base64Binary" />
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required" />
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="error" type="errorType" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- progress -->
  <xs:element name="progress" type="progressType"/>
</xs:schema>

```

Пример 1:

Импорт валидного контекста

Запрос:

http://192.168.23.34:9999/commands/modificators_context_import


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- contextType-->
  <xs:complexType name="contextType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="contexts">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="context" type="contextType" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/modifiers_context_list

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_modifiers_context_list.xsd">
  <request domain="biysk.local"/>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_modific
ators_context_list.xsd">
  <contexts>
    <context name="mod_smg"/>
    <context name="default_modifiers"/>
    <context name="mod_1"/>
  </contexts>
</out>
```

hc_modifiers_context_show

Команда предназначена для получения контекста модификации номеров по имени из определенного домена.

Аналог команды в CoCon:

cluster/storage/<STORAGE>/modifiers/show

domain/<DOMAIN>/modifiers/show

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/modifiers_context_show

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="context" type="xs:base64Binary"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="context" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

В поле context в результате в кодировке base64 закодирован контекст модификации (в формате XML). XSD схема, валидирующая контексты модификации:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="ecss_acp_common.xsd"/>
  <!-- booleanType -->
  <xs:simpleType name="booleanType">
    <xs:restriction base="xs:boolean"/>
  </xs:simpleType>
  <!-- ruleType -->
  <xs:complexType name="ruleType">
    <xs:sequence>
      <xs:element name="conditions" minOccurs="0">
        <xs:complexType>
          <xs:all>
            <xs:element name="cgpn" type="cgpnType" minOccurs="0"/>
            <xs:element name="cdpn" type="cdpnType" minOccurs="0"/>
            <xs:element name="rgn" type="rgnConditionType" minOccurs="0"/>
            <xs:element name="ocdpn" type="ocdpnConditionType" minOccurs="0"/>
            <xs:element name="cn" type="cnConditionType" minOccurs="0"/>
            <xs:element name="time" type="valueType" minOccurs="0"/>
            <xs:element name="date" type="valueType" minOccurs="0"/>
            <xs:element name="weekday" type="valueType" minOccurs="0"/>
            <xs:element name="tag" type="valueType" minOccurs="0"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
      <xs:element name="actions" minOccurs="0">
        <xs:complexType>
          <xs:all>
            <xs:element name="cgpn" type="cgpnType" minOccurs="0"/>
            <xs:element name="cdpn" type="cdpnType" minOccurs="0"/>
            <xs:element name="rgn" type="rgnActionType" minOccurs="0"/>
            <xs:element name="ocdpn" type="ocdpnActionType" minOccurs="0"/>
            <xs:element name="cn" type="cnActionType" minOccurs="0"/>
            <xs:element name="empty_rgn" type="nullType" minOccurs="0">
              <xs:annotation><xs:documentation xml:lang="en">Remove routing
number.</xs:documentation></xs:annotation>
            </xs:element>
            <xs:element name="empty_ocdpn" type="nullType" minOccurs="0">
              <xs:annotation><xs:documentation xml:lang="en">Remove original called
number.</xs:documentation></xs:annotation>
            </xs:element>
            <xs:element name="empty_cn" type="nullType" minOccurs="0">
              <xs:annotation><xs:documentation xml:lang="en">Remove connected
number.</xs:documentation></xs:annotation>
            </xs:element>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

```

```

    <xs:element name="result">
      <xs:complexType>
        <xs:choice>
          <xs:element name="finish" type="finishResultType"/>
          <xs:element name="error" type="errorResultType"/>
          <xs:element name="continue" type="continueResultType"/>
        </xs:choice>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="description" type="xs:string" use="optional"/>
</xs:complexType>
<!-- modifiersType -->
<xs:complexType name="modifiersType">
  <xs:all>
    <xs:element name="in" minOccurs="0" maxOccurs="1">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="rule" type="ruleType" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="out" minOccurs="0" maxOccurs="1">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="rule" type="ruleType" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:all>
  <xs:attribute name="name" type="xs:string" use="required">
    <xs:annotation>
      <xs:documentation>The name of the current modifiers.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="description" type="xs:string" use="optional">
    <xs:annotation>
      <xs:documentation>Text description of the current modifiers.</xs:documentation
>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
<!-- adaptationType -->
<xs:complexType name="adaptationType">
  <xs:sequence>
    <xs:element name="rule" type="ruleType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required">
    <xs:annotation>

```



```

        <xs:documentation>The name of the current adaptation.</xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="description" type="xs:string" use="optional">
    <xs:annotation>
        <xs:documentation>Text description of the current adaptation.</xs:documentation>
    </xs:annotation>
</xs:attribute>
</xs:complexType>
<!-- modifiers -->
<xs:element name="modifiers" type="modifiersType"/>
<!-- adaptation -->
<xs:element name="adaptation" type="adaptationType"/>
<!-- cdpnType -->
<xs:complexType name="cdpnType">
    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="nai" type="naiType"/>
    <xs:attribute name="incomplete" type="xs:boolean"/>
    <xs:attribute name="inni" type="inniType"/>
    <xs:attribute name="npi" type="npiType"/>
    <xs:attribute name="ni" type="niType"/>
    <xs:attribute name="category" type="categoryType"/>
</xs:complexType>
<!-- cgpnType -->
<xs:complexType name="cgpnType">
    <xs:attribute name="nai" type="naiType"/>
    <xs:attribute name="incomplete" type="xs:boolean"/>
    <xs:attribute name="npi" type="npiType"/>
    <xs:attribute name="apri" type="apriType"/>
    <xs:attribute name="screening" type="screeningType"/>
    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="ni" type="niType"/>
</xs:complexType>
<!-- rgnConditionType -->
<xs:complexType name="rgnConditionType">
    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="nai" type="naiType"/>
    <xs:attribute name="incomplete" type="xs:boolean"/>
    <xs:attribute name="apri" type="apriType"/>
    <xs:attribute name="npi" type="npiType"/>
    <xs:attribute name="ni" type="niType"/>
    <xs:attribute name="empty" type="booleanType">
        <xs:annotation>
            <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>
<!-- ocdpnConditionType -->
<xs:complexType name="ocdpnConditionType">

```

```

<xs:attribute name="digits" type="xs:string"/>
<xs:attribute name="nai" type="naiType"/>
<xs:attribute name="ni" type="niType"/>
<xs:attribute name="npi" type="npiType"/>
<xs:attribute name="apri" type="apriType"/>
<xs:attribute name="category" type="categoryType"/>
<xs:attribute name="incomplete" type="xs:boolean"/>
<xs:attribute name="empty" type="booleanType">
  <xs:annotation>
    <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
  </xs:annotation>
</xs:attribute>
</xs:complexType>
<!-- cnConditionType -->
<xs:complexType name="cnConditionType">
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="apri" type="apriType"/>
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="ni" type="niType"/>
  <xs:attribute name="empty" type="booleanType">
    <xs:annotation>
      <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
<!-- rgnActionType -->
<xs:complexType name="rgnActionType">
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
  <xs:attribute name="apri" type="apriType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="ni" type="niType"/>
</xs:complexType>
<!-- ocdpnActionType -->
<xs:complexType name="ocdpnActionType">
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="ni" type="niType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="apri" type="apriType"/>
  <xs:attribute name="category" type="categoryType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
</xs:complexType>
<!-- cnActionType -->
<xs:complexType name="cnActionType">
  <xs:attribute name="nai" type="naiType"/>

```

```

<xs:attribute name="npi" type="npiType"/>
<xs:attribute name="apri" type="apriType"/>
<xs:attribute name="digits" type="xs:string"/>
<xs:attribute name="ni" type="niType"/>
<xs:attribute name="empty" type="booleanType">
  <xs:annotation>
    <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
  </xs:annotation>
</xs:attribute>
</xs:complexType>
<!-- valueType -->
<xs:complexType name="valueType">
  <xs:attribute name="value" type="xs:string" use="required"/>
</xs:complexType>
<!-- nullType -->
<xs:complexType name="nullType"/>
<!-- Result types -->
<!-- finishResultType -->
<xs:complexType name="finishResultType"/>
<!-- errorResultType -->
<xs:complexType name="errorResultType">
  <xs:attribute name="acp_cause" type="xs:string" use="optional"/>
  <xs:attribute name="isup_cause" type="xs:string" use="optional"/>
  <xs:attribute name="description" type="xs:string" use="optional"/>
</xs:complexType>
<!-- continueResultTypeType -->
<xs:simpleType name="continueResultTypeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="start"/>
    <xs:enumeration value="next"/>
  </xs:restriction>
</xs:simpleType>
<!-- continueResultType -->
<xs:complexType name="continueResultType">
  <xs:attribute name="tag" type="xs:string" use="optional"/>
  <xs:attribute name="type" type="continueResultTypeType" default="start" use="optional"/>
</xs:complexType>
</xs:schema>

```

Пример 1:

Запрос существующего контекста

Запрос:

http://192.168.1.21:9999/commands/modifiers_context_show


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- categoryType -->
  <xs:simpleType name="categoryType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="unknownAtThisTime"/>
      <xs:enumeration value="operatorFrench"/>
      <xs:enumeration value="operatorEnglish">
        <xs:annotation>
          <xs:documentation xml:lang="en">Obsolete value. Correct value is
operatorEnglish</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="operatorEnglish"/>
      <xs:enumeration value="operatorGerman"/>
      <xs:enumeration value="operatorRussian"/>
      <xs:enumeration value="operatorSpanish"/>
      <xs:enumeration value="reserved"/>
      <xs:enumeration value="ordinarySubscriber"/>
      <xs:enumeration value="subscriberWithPriority"/>
      <xs:enumeration value="dataCall"/>
      <xs:enumeration value="testCall"/>
      <xs:enumeration value="spare"/>
      <xs:enumeration value="payphone"/>
      <xs:enumeration value="category0"/>
      <xs:enumeration value="hotelsSubscriber"/>
      <xs:enumeration value="freeSubscriber"/>
      <xs:enumeration value="paidSubscriber"/>
      <xs:enumeration value="localSubscriber"/>
      <xs:enumeration value="localTaksofon"/>
      <xs:enumeration value="autoCallI"/>
      <xs:enumeration value="semiautoCallI"/>
      <xs:enumeration value="autoCallII"/>
      <xs:enumeration value="semiautoCallII"/>
      <xs:enumeration value="autoCallIII"/>
      <xs:enumeration value="semiautoCallIII"/>
      <xs:enumeration value="autoCallIV"/>
      <xs:enumeration value="semiautoCallIV"/>
      <xs:enumeration value="0"/>
      <xs:enumeration value="1"/>
      <xs:enumeration value="2"/>
      <xs:enumeration value="3"/>
      <xs:enumeration value="4"/>
      <xs:enumeration value="5"/>
      <xs:enumeration value="6"/>
      <xs:enumeration value="7"/>
      <xs:enumeration value="8"/>
      <xs:enumeration value="9"/>
    </xs:restriction>
  </xs:simpleType>

```

```
<xs:enumeration value="10"/>
<xs:enumeration value="11"/>
<xs:enumeration value="12"/>
<xs:enumeration value="13"/>
<xs:enumeration value="14"/>
<xs:enumeration value="15"/>
<xs:enumeration value="16"/>
<xs:enumeration value="17"/>
<xs:enumeration value="18"/>
<xs:enumeration value="19"/>
<xs:enumeration value="20"/>
<xs:enumeration value="21"/>
<xs:enumeration value="22"/>
<xs:enumeration value="23"/>
<xs:enumeration value="24"/>
<xs:enumeration value="25"/>
<xs:enumeration value="26"/>
<xs:enumeration value="27"/>
<xs:enumeration value="28"/>
<xs:enumeration value="29"/>
<xs:enumeration value="30"/>
<xs:enumeration value="31"/>
<xs:enumeration value="32"/>
<xs:enumeration value="33"/>
<xs:enumeration value="34"/>
<xs:enumeration value="35"/>
<xs:enumeration value="36"/>
<xs:enumeration value="37"/>
<xs:enumeration value="38"/>
<xs:enumeration value="39"/>
<xs:enumeration value="40"/>
<xs:enumeration value="41"/>
<xs:enumeration value="42"/>
<xs:enumeration value="43"/>
<xs:enumeration value="44"/>
<xs:enumeration value="45"/>
<xs:enumeration value="46"/>
<xs:enumeration value="47"/>
<xs:enumeration value="48"/>
<xs:enumeration value="49"/>
<xs:enumeration value="50"/>
<xs:enumeration value="51"/>
<xs:enumeration value="52"/>
<xs:enumeration value="53"/>
<xs:enumeration value="54"/>
<xs:enumeration value="55"/>
<xs:enumeration value="56"/>
<xs:enumeration value="57"/>
<xs:enumeration value="58"/>
<xs:enumeration value="59"/>
```

```
<xs:enumeration value="60"/>
<xs:enumeration value="61"/>
<xs:enumeration value="62"/>
<xs:enumeration value="63"/>
<xs:enumeration value="64"/>
<xs:enumeration value="65"/>
<xs:enumeration value="66"/>
<xs:enumeration value="67"/>
<xs:enumeration value="68"/>
<xs:enumeration value="69"/>
<xs:enumeration value="70"/>
<xs:enumeration value="71"/>
<xs:enumeration value="72"/>
<xs:enumeration value="73"/>
<xs:enumeration value="74"/>
<xs:enumeration value="75"/>
<xs:enumeration value="76"/>
<xs:enumeration value="77"/>
<xs:enumeration value="78"/>
<xs:enumeration value="79"/>
<xs:enumeration value="80"/>
<xs:enumeration value="81"/>
<xs:enumeration value="82"/>
<xs:enumeration value="83"/>
<xs:enumeration value="84"/>
<xs:enumeration value="85"/>
<xs:enumeration value="86"/>
<xs:enumeration value="87"/>
<xs:enumeration value="88"/>
<xs:enumeration value="89"/>
<xs:enumeration value="90"/>
<xs:enumeration value="91"/>
<xs:enumeration value="92"/>
<xs:enumeration value="93"/>
<xs:enumeration value="94"/>
<xs:enumeration value="95"/>
<xs:enumeration value="96"/>
<xs:enumeration value="97"/>
<xs:enumeration value="98"/>
<xs:enumeration value="99"/>
<xs:enumeration value="100"/>
<xs:enumeration value="101"/>
<xs:enumeration value="102"/>
<xs:enumeration value="103"/>
<xs:enumeration value="104"/>
<xs:enumeration value="105"/>
<xs:enumeration value="106"/>
<xs:enumeration value="107"/>
<xs:enumeration value="108"/>
<xs:enumeration value="109"/>
```

```
<xs:enumeration value="110"/>
<xs:enumeration value="111"/>
<xs:enumeration value="112"/>
<xs:enumeration value="113"/>
<xs:enumeration value="114"/>
<xs:enumeration value="115"/>
<xs:enumeration value="116"/>
<xs:enumeration value="117"/>
<xs:enumeration value="118"/>
<xs:enumeration value="119"/>
<xs:enumeration value="120"/>
<xs:enumeration value="121"/>
<xs:enumeration value="122"/>
<xs:enumeration value="123"/>
<xs:enumeration value="124"/>
<xs:enumeration value="125"/>
<xs:enumeration value="126"/>
<xs:enumeration value="127"/>
<xs:enumeration value="128"/>
<xs:enumeration value="129"/>
<xs:enumeration value="130"/>
<xs:enumeration value="131"/>
<xs:enumeration value="132"/>
<xs:enumeration value="133"/>
<xs:enumeration value="134"/>
<xs:enumeration value="135"/>
<xs:enumeration value="136"/>
<xs:enumeration value="137"/>
<xs:enumeration value="138"/>
<xs:enumeration value="139"/>
<xs:enumeration value="140"/>
<xs:enumeration value="141"/>
<xs:enumeration value="142"/>
<xs:enumeration value="143"/>
<xs:enumeration value="144"/>
<xs:enumeration value="145"/>
<xs:enumeration value="146"/>
<xs:enumeration value="147"/>
<xs:enumeration value="148"/>
<xs:enumeration value="149"/>
<xs:enumeration value="150"/>
<xs:enumeration value="151"/>
<xs:enumeration value="152"/>
<xs:enumeration value="153"/>
<xs:enumeration value="154"/>
<xs:enumeration value="155"/>
<xs:enumeration value="156"/>
<xs:enumeration value="157"/>
<xs:enumeration value="158"/>
<xs:enumeration value="159"/>
```



```
<xs:enumeration value="160"/>
<xs:enumeration value="161"/>
<xs:enumeration value="162"/>
<xs:enumeration value="163"/>
<xs:enumeration value="164"/>
<xs:enumeration value="165"/>
<xs:enumeration value="166"/>
<xs:enumeration value="167"/>
<xs:enumeration value="168"/>
<xs:enumeration value="169"/>
<xs:enumeration value="170"/>
<xs:enumeration value="171"/>
<xs:enumeration value="172"/>
<xs:enumeration value="173"/>
<xs:enumeration value="174"/>
<xs:enumeration value="175"/>
<xs:enumeration value="176"/>
<xs:enumeration value="177"/>
<xs:enumeration value="178"/>
<xs:enumeration value="179"/>
<xs:enumeration value="180"/>
<xs:enumeration value="181"/>
<xs:enumeration value="182"/>
<xs:enumeration value="183"/>
<xs:enumeration value="184"/>
<xs:enumeration value="185"/>
<xs:enumeration value="186"/>
<xs:enumeration value="187"/>
<xs:enumeration value="188"/>
<xs:enumeration value="189"/>
<xs:enumeration value="190"/>
<xs:enumeration value="191"/>
<xs:enumeration value="192"/>
<xs:enumeration value="193"/>
<xs:enumeration value="194"/>
<xs:enumeration value="195"/>
<xs:enumeration value="196"/>
<xs:enumeration value="197"/>
<xs:enumeration value="198"/>
<xs:enumeration value="199"/>
<xs:enumeration value="200"/>
<xs:enumeration value="201"/>
<xs:enumeration value="202"/>
<xs:enumeration value="203"/>
<xs:enumeration value="204"/>
<xs:enumeration value="205"/>
<xs:enumeration value="206"/>
<xs:enumeration value="207"/>
<xs:enumeration value="208"/>
<xs:enumeration value="209"/>
```

```
<xs:enumeration value="210"/>
<xs:enumeration value="211"/>
<xs:enumeration value="212"/>
<xs:enumeration value="213"/>
<xs:enumeration value="214"/>
<xs:enumeration value="215"/>
<xs:enumeration value="216"/>
<xs:enumeration value="217"/>
<xs:enumeration value="218"/>
<xs:enumeration value="219"/>
<xs:enumeration value="220"/>
<xs:enumeration value="221"/>
<xs:enumeration value="222"/>
<xs:enumeration value="223"/>
<xs:enumeration value="224"/>
<xs:enumeration value="225"/>
<xs:enumeration value="226"/>
<xs:enumeration value="227"/>
<xs:enumeration value="228"/>
<xs:enumeration value="229"/>
<xs:enumeration value="230"/>
<xs:enumeration value="231"/>
<xs:enumeration value="232"/>
<xs:enumeration value="233"/>
<xs:enumeration value="234"/>
<xs:enumeration value="235"/>
<xs:enumeration value="236"/>
<xs:enumeration value="237"/>
<xs:enumeration value="238"/>
<xs:enumeration value="239"/>
<xs:enumeration value="240"/>
<xs:enumeration value="241"/>
<xs:enumeration value="242"/>
<xs:enumeration value="243"/>
<xs:enumeration value="244"/>
<xs:enumeration value="245"/>
<xs:enumeration value="246"/>
<xs:enumeration value="247"/>
<xs:enumeration value="248"/>
<xs:enumeration value="249"/>
<xs:enumeration value="250"/>
<xs:enumeration value="251"/>
<xs:enumeration value="252"/>
<xs:enumeration value="253"/>
<xs:enumeration value="254"/>
<xs:enumeration value="255"/>
</xs:restriction>
</xs:simpleType>
<!-- naiType -->
<xs:simpleType name="naiType">
```

```

    <xs:restriction base="xs:string">
      <xs:enumeration value="spare"/>
      <xs:enumeration value="subscriberNumber"/>
      <xs:enumeration value="unknown"/>
      <xs:enumeration value="nationalNumber"/>
      <xs:enumeration value="internationalNumber"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- inniType -->
  <xs:simpleType name="inniType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="routingToInternalNumberAllowed"/>
      <xs:enumeration value="routingToInternalNumberNotAllowed"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- npiType -->
  <xs:simpleType name="npiType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="spare"/>
      <xs:enumeration value="isdntelephony"/>
      <xs:enumeration value="dataNumberingPlan"/>
      <xs:enumeration value="telexNumberingPlan"/>
      <xs:enumeration value="reserved1"/>
      <xs:enumeration value="reserved2"/>
      <xs:enumeration value="reserved3"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- niType -->
  <xs:simpleType name="niType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="private"/>
      <xs:enumeration value="local"/>
      <xs:enumeration value="zone"/>
      <xs:enumeration value="intercity"/>
      <xs:enumeration value="international"/>
      <xs:enumeration value="emergency"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- apriType -->
  <xs:simpleType name="apriType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="presentationAllowed"/>
      <xs:enumeration value="presentationRestricted"/>
      <xs:enumeration value="addressNotAvailable"/>
      <xs:enumeration value="spare"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- screeningType -->
  <xs:simpleType name="screeningType">
    <xs:restriction base="xs:string">

```

```

        <xs:enumeration value="userProvidedNotVerified"/>
        <xs:enumeration value="userProvidedVerifiedAndPassed"/>
        <xs:enumeration value="userProvidedVerifiedAndFailed"/>
        <xs:enumeration value="networkProvided"/>
    </xs:restriction>
</xs:simpleType>
<!-- booleanType -->
<xs:simpleType name="booleanType">
    <xs:restriction base="xs:boolean"/>
</xs:simpleType>
<!-- ruleType -->
<xs:complexType name="ruleType">
    <xs:sequence>
        <xs:element name="conditions" minOccurs="0">
            <xs:complexType>
                <xs:all>
                    <xs:element name="cgpn" type="cgpnType" minOccurs="0"/>
                    <xs:element name="cdpn" type="cdpnType" minOccurs="0"/>
                    <xs:element name="rgn" type="rgnConditionType" minOccurs="0"/>
                    <xs:element name="rnn" type="rnnConditionType" minOccurs="0"/>
                    <xs:element name="ocdpn" type="ocdpnConditionType" minOccurs="0"/>
                    <xs:element name="cn" type="cnConditionType" minOccurs="0"/>
                    <xs:element name="time" type="valueType" minOccurs="0"/>
                    <xs:element name="date" type="valueType" minOccurs="0"/>
                    <xs:element name="weekday" type="valueType" minOccurs="0"/>
                    <xs:element name="tag" type="valueType" minOccurs="0"/>
                </xs:all>
            </xs:complexType>
        </xs:element>
        <xs:element name="actions" minOccurs="0">
            <xs:complexType>
                <xs:all>
                    <xs:element name="cgpn" type="cgpnType" minOccurs="0"/>
                    <xs:element name="cdpn" type="cdpnType" minOccurs="0"/>
                    <xs:element name="rgn" type="rgnActionType" minOccurs="0"/>
                    <xs:element name="rnn" type="rnnActionType" minOccurs="0"/>
                    <xs:element name="ocdpn" type="ocdpnActionType" minOccurs="0"/>
                    <xs:element name="cn" type="cnActionType" minOccurs="0"/>
                    <xs:element name="empty_rgn" type="nullType" minOccurs="0">
                        <xs:annotation><xs:documentation xml:lang="en">Remove routing
number.</xs:documentation></xs:annotation>
                    </xs:element>
                    <xs:element name="empty_ocdpn" type="nullType" minOccurs="0">
                        <xs:annotation><xs:documentation xml:lang="en">Remove original called
number.</xs:documentation></xs:annotation>
                    </xs:element>
                    <xs:element name="empty_cn" type="nullType" minOccurs="0">
                        <xs:annotation><xs:documentation xml:lang="en">Remove connected
number.</xs:documentation></xs:annotation>
                    </xs:element>
                </xs:all>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>

```

```

        </xs:all>
    </xs:complexType>
</xs:element>
<xs:element name="result">
    <xs:complexType>
        <xs:choice>
            <xs:element name="finish" type="finishResultType"/>
            <xs:element name="error" type="errorResultType"/>
            <xs:element name="next" type="nextResultType"/>
            <xs:element name="continue" type="continueResultType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="name" type="xs:string" use="required"/>
<xs:attribute name="description" type="xs:string" use="optional"/>
</xs:complexType>
<!-- modifiersType -->
<xs:complexType name="modifiersType">
    <xs:all>
        <xs:element name="in" minOccurs="0" maxOccurs="1">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="rule" type="ruleType" maxOccurs="unbounded"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="out" minOccurs="0" maxOccurs="1">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="rule" type="ruleType" maxOccurs="unbounded"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:all>
    <xs:attribute name="name" type="xs:string" use="required">
        <xs:annotation>
            <xs:documentation>The name of the current modifiers.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="description" type="xs:string" use="optional">
        <xs:annotation>
            <xs:documentation>Text description of the current modifiers.</
xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>
<!-- adaptationType -->
<xs:complexType name="adaptationType">
    <xs:sequence>

```

```

        <xs:element name="rule" type="ruleType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required">
        <xs:annotation>
            <xs:documentation>The name of the current adaptation.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="description" type="xs:string" use="optional">
        <xs:annotation>
            <xs:documentation>Text description of the current adaptation.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>
<!-- modifiers -->
<xs:element name="modifiers" type="modifiersType"/>
<!-- adaptation -->
<xs:element name="adaptation" type="adaptationType"/>
<!-- cdpnType -->
<xs:complexType name="cdpnType">
    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="nai" type="naiType"/>
    <xs:attribute name="incomplete" type="xs:boolean"/>
    <xs:attribute name="inni" type="inniType"/>
    <xs:attribute name="npi" type="npiType"/>
    <xs:attribute name="ni" type="niType"/>
    <xs:attribute name="category" type="categoryType"/>
    <xs:attribute name="in_list" type="xs:string"/>
    <xs:attribute name="display_name" type="xs:string"/>
    <xs:attribute name="city" type="xs:string"/>
    <xs:attribute name="region" type="xs:string"/>
    <xs:attribute name="operator" type="xs:string"/>
</xs:complexType>
<!-- cgpnType -->
<xs:complexType name="cgpnType">
    <xs:attribute name="nai" type="naiType"/>
    <xs:attribute name="incomplete" type="xs:boolean"/>
    <xs:attribute name="npi" type="npiType"/>
    <xs:attribute name="apri" type="apriType"/>
    <xs:attribute name="screening" type="screeningType"/>
    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="ni" type="niType"/>
    <xs:attribute name="in_list" type="xs:string"/>
    <xs:attribute name="display_name" type="xs:string"/>
    <xs:attribute name="transit_display_name" type="booleanType"/>
    <xs:attribute name="caller_id" type="xs:string"/>
    <xs:attribute name="city" type="xs:string"/>
    <xs:attribute name="region" type="xs:string"/>
    <xs:attribute name="operator" type="xs:string"/>
</xs:complexType>
<!-- rgnConditionType -->

```

```

<xs:complexType name="rgnConditionType">
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
  <xs:attribute name="apri" type="apriType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="ni" type="niType"/>
  <xs:attribute name="in_list" type="xs:string"/>
  <xs:attribute name="empty" type="booleanType">
    <xs:annotation>
      <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
<!-- rnnConditionType -->
<xs:complexType name="rnnConditionType">
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
  <xs:attribute name="inni" type="inniType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="ni" type="niType"/>
  <xs:attribute name="in_list" type="xs:string"/>
  <xs:attribute name="empty" type="booleanType">
    <xs:annotation>
      <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
<!-- ocdpnConditionType -->
<xs:complexType name="ocdpnConditionType">
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="ni" type="niType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="apri" type="apriType"/>
  <xs:attribute name="category" type="categoryType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
  <xs:attribute name="in_list" type="xs:string"/>
  <xs:attribute name="empty" type="booleanType">
    <xs:annotation>
      <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
<!-- cnConditionType -->
<xs:complexType name="cnConditionType">

```

```

<xs:attribute name="nai" type="naiType"/>
<xs:attribute name="npi" type="npiType"/>
<xs:attribute name="apri" type="apriType"/>
<xs:attribute name="digits" type="xs:string"/>
<xs:attribute name="ni" type="niType"/>
<xs:attribute name="in_list" type="xs:string"/>
<xs:attribute name="empty" type="booleanType">
  <xs:annotation>
    <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
  </xs:annotation>
</xs:attribute>
</xs:complexType>
<!-- rgnActionType -->
<xs:complexType name="rgnActionType">
  <xs:attribute name="display_name" type="xs:string"/>
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
  <xs:attribute name="apri" type="apriType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="ni" type="niType"/>
</xs:complexType>
<!-- rnnActionType -->
<xs:complexType name="rnnActionType">
  <xs:attribute name="display_name" type="xs:string"/>
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
  <xs:attribute name="inni" type="inniType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="ni" type="niType"/>
</xs:complexType>
<!-- ocdpnActionType -->
<xs:complexType name="ocdpnActionType">
  <xs:attribute name="display_name" type="xs:string"/>
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="ni" type="niType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="apri" type="apriType"/>
  <xs:attribute name="category" type="categoryType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
</xs:complexType>
<!-- cnActionType -->
<xs:complexType name="cnActionType">
  <xs:attribute name="display_name" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="apri" type="apriType"/>

```



```

    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="ni" type="niType"/>
    <xs:attribute name="caller_id" type="xs:string"/>
    <xs:attribute name="transit_display_name" type="booleanType"/>
</xs:complexType>
<!-- valueType -->
<xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:string" use="required"/>
</xs:complexType>
<!-- nullType -->
<xs:complexType name="nullType"/>
<!-- Result types -->
<!-- finishResultType -->
<xs:complexType name="finishResultType"/>
<!-- errorResultType -->
<xs:complexType name="errorResultType">
    <xs:attribute name="acp_cause" type="xs:string" use="optional"/>
    <xs:attribute name="isup_cause" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
</xs:complexType>
<!-- continueResultTypeType -->
<xs:simpleType name="continueResultTypeType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="start"/>
        <xs:enumeration value="next"/>
    </xs:restriction>
</xs:simpleType>
<!-- nextResultType -->
<xs:complexType name="nextResultType">
    <xs:attribute name="tag" type="xs:string" use="optional"/>
</xs:complexType>
<!-- continueResultType -->
<xs:complexType name="continueResultType">
    <xs:attribute name="tag" type="xs:string" use="optional"/>
    <xs:attribute name="type" type="continueResultTypeType" default="start" use="optional"/>
</xs:complexType>
</xs:schema>

```

Нс_msr_timers_info - Информация о таймерах MSR

[Нс_msr_timers_info](#)

Шаблон URL запроса

Аналог команды в CoSop:

domain/<DOMAIN>/timers/msr/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/msr_timers_info

Коды ответа

200 - успех;

404 - ошибка.

XML-схема

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="msr_timers_info">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show msr timers properties
      Copyright (c) 2019, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <xs:complexType name="msrTimersOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="bh_timeout" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="rfc_4028_use_se" type="nonEmptyTokenType" use="optional" />
        <xs:attribute name="rfc_4028_min_se" type="nonEmptyTokenType" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainInfoInType"
                minOccurs="1" maxOccurs="1" />
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainInfoOutType"
                minOccurs="1" maxOccurs="1" />
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- out -->

```

```

        </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
    </xs:choice>
</xs:complexType>
</xs:element>
<!-- out -->

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/msr_timers_info

```

<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="msr_timers_info.xsd">
  <request>
    <domain_options domain="biysk.local" />
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="msr_timers_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="msrTimersOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="msrTimersOptionsType" rfc_4028_use_se="1800000"
rfc_4028_min_se="900000" bh_timeout="200"/>
      </domain>
    </domain_options>
  </result>
</out>

```

hc_node_last_errors - Список N последних error-логов

Команда возвращает список последних ошибок узла (количество по умолчанию: 50).

Метод http запроса

POST

Шаблон http запроса

http://host:port/commands/node_last_errors

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="count" type="xs:integer" use="optional" default="50"/>
  </xs:complexType>

  <xs:complexType name="logType">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="time" type="xs:string" use="required"/>
    <xs:attribute name="module" type="xs:string" use="required"/>
    <xs:attribute name="line" type="xs:string" use="required"/>
    <xs:attribute name="msg" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="node" type="nodeType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="log" type="logType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример 1:

Вывод всех ошибок на core1@ecss1

Запрос: http://192.168.1.21:9999/commands/node_last_errors

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_node_last_errors.xsd">
  <node name="core1@ecss1" />
</in>
```

Ответ: 200;

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_node_last_errors.xsd">
  <log id="0" module="oct_ss_versioner" line="54"
    msg="Supplementary service zone_page release 12 version, but installed 10 version."
    time="07.04.2020 15:54:24"/>
  <log id="1" module="oct_ss_versioner" line="54"
    msg="Supplementary service voicemail release 14 version, but installed 13 version."
    time="07.04.2020 15:54:24"/>
  <log id="2" module="oct_ss_versioner" line="54"
    msg="Supplementary service auto_redial release 3 version, but installed 2 version."
    time="07.04.2020 15:54:24"/>
  <log id="3" module="oct_ss_versioner" line="54"
    msg="Supplementary service blf release 4 version, but installed 3 version."
time="07.04.2020 15:54:24"/>
  <log id="4" module="oct_ss_versioner" line="54"
    msg="Supplementary service presence release 3 version, but installed 2 version."
time="07.04.2020 15:54:24"/>
  <log id="5" module="oct_ss_versioner" line="54"
    msg="Supplementary service message release 3 version, but installed 2 version."
time="07.04.2020 15:54:24"/>
  <log id="6" module="oct_ss_versioner" line="54"
    msg="Supplementary service voice_page release 10 version, but installed 9 version."
    time="07.04.2020 15:54:24"/>
  <log id="7" module="oct_ss_versioner" line="54"
    msg="Supplementary service cf_aon release 10 version, but installed 9 version."
time="07.04.2020 15:54:24"/>
  <log id="8" module="oct_ss_versioner" line="54"
    msg="Supplementary service direct_call release 3 version, but installed 2 version."
    time="07.04.2020 15:54:24"/>
  <log id="9" module="oct_ss_versioner" line="54"
    msg="Supplementary service conference release 10 version, but installed 9 version."
    time="07.04.2020 15:54:24"/>
  <log id="10" module="oct_ss_versioner" line="54"
    msg="Supplementary service cft release 15 version, but installed 14 version."
time="07.04.2020 15:54:24"/>
  <log id="11" module="oct_ss_versioner" line="54"
    msg="Supplementary service chunt release 8 version, but installed 7 version."
time="07.04.2020 15:54:24"/>
  <log id="12" module="oct_ss_versioner" line="54"
    msg="Supplementary service cfnr_type2 release 12 version, but installed 11 version."
    time="07.04.2020 15:54:24"/>
  <log id="13" module="oct_ss_versioner" line="54"
    msg="Supplementary service cfnr release 12 version, but installed 11 version."
time="07.04.2020 15:54:24"/>
  <log id="14" module="oct_ss_versioner" line="54"
    msg="Supplementary service alarm release 9 version, but installed 8 version."
time="07.04.2020 15:54:24"/>
  <log id="15" module="oct_ss_versioner" line="54"
```

```

        msg="Supplementary service auto_redial_with_callback release 7 version, but installed 6
version."
        time="07.04.2020 15:54:24"/>
    <log id="16" module="oct_ss_versioner" line="54"
        msg="Supplementary service callback release 6 version, but installed 5 version."
time="07.04.2020 15:54:24"/>
    <log id="17" module="oct_ss_versioner" line="54"
        msg="Supplementary service follow_me_no_response release 9 version, but installed 8
version."
        time="07.04.2020 15:54:25"/>
    <log id="18" module="oct_ss_versioner" line="54"
        msg="Supplementary service find_me_no_response release 7 version, but installed 6
version."
        time="07.04.2020 15:54:25"/>
    <log id="19" module="oct_ss_versioner" line="54"
        msg="Supplementary service tsmn_request release 4 version, but installed 3 version."
        time="07.04.2020 15:54:25"/>
</out>

```

Пример 2:

Вывод последней ошибки на core1@ecss1

Запрос: http://192.168.1.21:9999/commands/node_last_errors

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_node_last_errors.xsd">
    <node name="core1@ecss1" count="1"/>
</in>

```

Ответ 200;

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_node_last_errors.xsd">
    <log id="19" module="oct_ss_versioner" line="54" msg="Supplementary service tsmn_request
release 4 version, but installed 3 version." time="07.04.2020 15:54:25"/>
</out>

```

Нс_node_rps - Управление подсистемой сбора статистики и предупреждений ноды

- [Нс_node_rps_alarms](#)

- [Hc_node_rps_controls](#)
- [Hc_node_rps_events](#)
- [Hc_node_rps_logs](#)
- [Hc_node_rps_stats](#)
- [Hc_node_rps_status](#)

[Hc_node_rps_alarms](#)

Команда позволяет включить/выключить подсистемы аварийных сообщений ноды.

Аналог команды в CoCon:

`/node/<NODE>/rps/alarms`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/node_rps_alarms

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

`node_rps_onoff.xsd`

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name = "stateType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="on"/>
      <xs:enumeration value="off"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="state" type="stateType" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="node" type="nodeType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/node_rps_alarms

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="node_rps_onoff.xsd">
  <node name="ds1@ecss1" state="off" />
</in>

```

Ис_ node_rps_controls

Команда позволяет включить/выключить подсистему контроля ноды.

Аналог команды в CoCon:

/node/<NODE>/rps/controls

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/node_rps_controls

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа: - node_rps_onoff.xsd**Пример:**

Запрос: http://192.168.1.21:9999/commands/node_rps_controls

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="node_rps_onoff.xsd">
  <node name="ds1@ecss2" state="on" />
</in>
```

Ис_ node_rps_events

Команда позволяет включить/выключить подсистемы событий ноды.

Аналог команды в CoCon:

/node/<NODE>/rps/events

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/node_rps_events

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа: - node_rps_onoff.xsd**Пример:**

Запрос: http://192.168.1.21:9999/commands/node_rps_events

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="node_rps_onoff.xsd">
  <node name="ds1@ecss1" state="off" />
</in>
```

Нс_node_rps_logs

Команда позволяет включить/выключить подсистемы логов ноды.

Аналог команды в CoCon:

`/node/<NODE>/rps/logs`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/node_rps_logs

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа: - node_rps_onoff.xsd

Пример:

Запрос: http://192.168.1.21:9999/commands/node_rps_logs

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="node_rps_onoff.xsd">
  <node name="ds1@ecss2" state="off" />
</in>
```

Нс_node_rps_stats

Команда позволяет включить/выключить подсистемы статистики ноды.

Аналог команды в CoCon:

`/node/<NODE>/rps/stats`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/node_rps_stats

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа: - node_rps_onoff.xsd**Пример:**

Запрос: http://192.168.1.21:9999/commands/node_rps_stats

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="node_rps_onoff.xsd">
  <node name="ds1@ecss2" state="off" />
</in>
```

Ис_ node_rps_status

Команда позволяет вывести текущее состояние всех подсистем ноды.

Аналог команды в CoCon:

/node/<NODE>/rps/status

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/node_rps_status

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name = "stateType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="on"/>
      <xs:enumeration value="off"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="statusType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="state" type="stateType" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="node" type="nodeType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="status" type="statusType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/node_rps_status

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="node_rps_status.xsd">
  <node name="core1@ecss1" />
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="node_rps_status.xsd">
  <status name="alarms" state="on"/>
  <status name="controls" state="on"/>
  <status name="events" state="off"/>
  <status name="logs" state="off"/>
  <status name="statistics" state="on"/>
</out>
```

Hc_node - Управление нодами

- [Hc_node_memory_info](#)
- [Hc_node_application_info](#)
- [Hc_node_full_applications_info](#)
- [Hc_node_releases_info](#)
- [Hc_node_statistics_info](#)
- [Hc_node_system_info](#)
- [Hc_node_service](#)
- [Hc_node_show_cert](#)

[Hc_node_memory_info](#)

Команда позволяет просмотреть информацию об используемой памяти на ноде.

Аналог команды в CoCon:

`/node/<NODE>/info memory`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/node_memory_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="memoryOutType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:integer" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="node" type="nodeType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="memory" type="memoryOutType" minOccurs="0" maxOccurs="unbounded"
/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/node_memory_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_memory_info.xsd">
  <node name="ds1@ecss1" />
</in>

```

Ответ:


```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_memory_info.xsd">
  <memory name="total" value="150515304"/>
  <memory name="processes" value="32944552"/>
  <memory name="processes_used" value="32942752"/>
  <memory name="system" value="117570752"/>
  <memory name="atom" value="1796313"/>
  <memory name="atom_used" value="1785948"/>
  <memory name="binary" value="1404808"/>
  <memory name="code" value="42497375"/>
  <memory name="ets" value="20151392"/>
</out>
```

hc_node_application_info

Команда позволяет просмотреть информацию о приложениях (и его модулях), входящих в состав ноды.

Аналог команды в CoCon:

node/<NODE>/info applications

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/hc_node_application_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="application" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="moduleOutType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="vsn" type="xs:string" use="required"/>
    <xs:attribute name="date" type="xs:string" use="required"/>
    <xs:attribute name="commit" type="xs:string" use="required"/>
    <xs:attribute name="author" type="xs:string" use="required"/>
    <xs:attribute name="changes" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="applicationOutType">
    <xs:sequence>
      <xs:element name="module" type="moduleOutType" minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="state" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="vsn" type="xs:string" use="required"/>
    <xs:attribute name="crc" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
    <xs:attribute name="from" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="node" type="nodeType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="application" type="applicationOutType" minOccurs="0" maxOccurs="unbounded" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/node_application_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_application_info.xsd">
  <node application="chronica" name="core1@ecss1" />
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_application_info.xsd">
  <application state="true" name="chronica" vsn="3.14.0.165.393699" crc="97313144c50dd873cea95666c345f838aabaff12" description="Log Library" from="/usr/lib/ecss/ecss-core/lib/chronica-3.14.0.165.393699">
    <module name="chronica_app" vsn="[301922112467110895713685515611026976318]" date="2020/02/12 19:09:22" commit="70b13dbd71cde0f7415cb27c4d6ca25c70ab5ec9" author="Anton Ryabkov" changes="Wed Jan 29 10:49:45 2020 +0700"/>
    <module name="chronica_error_logger_handler" vsn="[208110506384551064294029479167702068891]" date="2020/02/12 19:09:23" commit="70b13dbd71cde0f7415cb27c4d6ca25c70ab5ec9" author="Anton Ryabkov" changes="Wed Jan 29 10:49:45 2020 +0700"/>
    <module name="chronica_n2o_log" vsn="[48290415293448581826704558131935557451]" date="2020/02/12 19:09:22" commit="70b13dbd71cde0f7415cb27c4d6ca25c70ab5ec9" author="Anton Ryabkov" changes="Wed Jan 29 10:49:45 2020 +0700"/>
    <module name="chronica_testing" vsn="[22335723895763387112073387528158927795]" date="2020/02/12 19:09:22" commit="70b13dbd71cde0f7415cb27c4d6ca25c70ab5ec9" author="Anton Ryabkov" changes="Wed Jan 29 10:49:45 2020 +0700"/>
  </application>
</out>
```

Нс_node_full_applications_info

Команда позволяет просмотреть информацию обо всех приложениях нод системы.

Аналог команды в CoCon:

/node/nodes-info applications

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/node_full_applications_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;

- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="applicationOutType">
    <xs:attribute name="state" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="vsn" type="xs:string" use="required"/>
    <xs:attribute name="crc" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
    <xs:attribute name="from" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="node" type="nodeType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="application" type="applicationOutType" minOccurs="0" maxOccurs="unbounded" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/node_full_applications_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_full_applications_info.xsd">
  <node name="sip1@ecss1" />
</in>
```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="node_full_applications_info.xsd">
  <application state="true" name="crypto" vsn="4.4.2"
  crc="f6c9a3875a33f9936708e2e58b103546f766b095" description="CRYPTO" from="/usr/lib/erlang/lib/
  crypto-4.4.2"/>
  <application state="true" name="sdp_lib" vsn="3.14.0.70231.537548"
  crc="fc85d9add6a643f0815096b1ee69592ab335b7c6" description="erlang library for sdp processing"
  from="/usr/lib/ecss/ecss-pa-sip/lib/sdp_lib-3.14.0.70231.537548"/>
  <application state="true" name="ecss_configuration_manager" vsn="3.14.0.70231.481099"
  crc="741ce50679bf36f2db87471487c6ffa6c6889c55" description="Universal Configuration Manager for
  ECSS" from="/usr/lib/ecss/ecss-pa-sip/lib/ecss_configuration_manager-3.14.0.70231.481099"/>
  <application state="true" name="ecss_pa_sip_common" vsn="3.14.0.70231.068633"
  crc="007387da70e8d572e61303be11053803bd1e6bd6" description="ECSS PA_SIP common library" from="/
  usr/lib/ecss/ecss-pa-sip/lib/ecss_pa_sip_common-3.14.0.70231.068633"/>
  <application state="true" name="mysql_driver" vsn="3.14.0.203.618242"
  crc="a0618f00a5ec46bee8fca312004e6a0998cf351b" description="Erlang MySQL driver" from="/usr/lib/
  ecss/ecss-pa-sip/lib/mysql_driver-3.14.0.203.618242"/>
  <application state="true" name="timer_container_lib" vsn="3.14.0.153.867866"
  crc="7967079842eeb7d0bc15f26abc1c4dcde9510b3e" description="Timer Container library" from="/usr/
  lib/ecss/ecss-pa-sip/lib/timer_container_lib-3.14.0.153.867866"/>
  <application state="true" name="acpadapter" vsn="3.14.0.70231.448922"
  crc="e22f6f26b12d38299dc498d5f02135bbe84b845" description="SIP(T) protocol adapter" from="/usr/
  lib/ecss/ecss-pa-sip/lib/acpadapter-3.14.0.70231.448922"/>
  <application state="true" name="chronica" vsn="3.14.0.165.393699"
  crc="57a1dee7f53e0b74a1896eb70e7b72fb79403fe5" description="Log Library" from="/usr/lib/ecss/
  ecss-pa-sip/lib/chronica-3.14.0.165.393699"/>
  <application state="true" name="isup_encoder" vsn="3.14.0.70231.487797"
  crc="a7bfaa9c639917bc9cb1215bae84398049ba367e" description="Realisation of ISUP encoder" from="/
  usr/lib/ecss/ecss-pa-sip/lib/isup_encoder-3.14.0.70231.487797"/>
  <application state="true" name="eldap" vsn="1.2.7"
  crc="50930607b192861e4decd405a87e6dfa02b202f1" description="Ldap api" from="/usr/lib/erlang/lib/
  eldap-1.2.7"/>
  <application state="true" name="ecss_starter" vsn="3.14.0.70231.401830"
  crc="0a75abf5e68586a415aa8f412ecd09c6245f503f" description="ECSS starter" from="/usr/lib/ecss/
  ecss-pa-sip/lib/ecss_starter-3.14.0.70231.401830"/>
  <application state="true" name="eep" vsn="1.0" crc="c2c551b069a89e389abe0757f55d9d769c56dc78"
  description="Erlang Easy Profiling (dbg:trace* to kcachegrind)" from="/usr/lib/ecss/ecss-pa-sip/
  lib/eep-1.0"/>
  <application state="true" name="kernel" vsn="6.3.1"
  crc="8ed7249b2a8c8b3037bafec369e7c3def2da7ad4d" description="ERTS CXC 138 10" from="/usr/lib/
  erlang/lib/kernel-6.3.1"/>
  <application state="true" name="rps_agent" vsn="3.14.0.70231.995766"
  crc="fe2d46bc222d1a6b9ba8a0baad45f24ec057c3ad" description="COCON Service Report processing
  system" from="/usr/lib/ecss/ecss-pa-sip/lib/rps_agent-3.14.0.70231.995766"/>
  <application state="true" name="mnesia" vsn="4.15.6"
  crc="dc6469a235534b432b846af68b2d816ad59a6f98" description="MNESIA CXC 138 12" from="/usr/lib/
  erlang/lib/mnesia-4.15.6"/>

```

```

    <application state="true" name="gen_netlink" vsn="0.3"
crc="406f08b56cd4a5ecaec7f218640e886cbe1927b8" description="Netlink socket toolkit" from="/usr/
lib/ecss/ecss-pa-sip/lib/gen_netlink-0.3"/>
    <application state="true" name="pt_scripts" vsn="3.14.0.103.511028"
crc="dc6c88608d8637c107984e29561ff59910547421" description="Parse transform scripts library"
from="/usr/lib/ecss/ecss-pa-sip/lib/pt_scripts-3.14.0.103.511028"/>
    <application state="true" name="notification_client" vsn="3.14.0.70231.255032"
crc="76c6239cfe04d72ecaf7748a26a8a35590fab247" description="Client notification" from="/usr/lib/
ecss/ecss-pa-sip/lib/notification_client-3.14.0.70231.255032"/>
    <application state="true" name="sorm_common_lib" vsn="3.14.0.70231.338138"
crc="231738cd52a338b501fd47e3b7fc69e4ad44bc9e" description="SORM common library" from="/usr/lib/
ecss/ecss-pa-sip/lib/sorm_common_lib-3.14.0.70231.338138"/>
    <application state="true" name="ecss_domain_options_lib" vsn="3.14.0.70231.830803"
crc="2c74d887f4caf7729be848890fd81a39a7c6b2f7" description="ECSS-10 domain options." from="/usr/
lib/ecss/ecss-pa-sip/lib/ecss_domain_options_lib-3.14.0.70231.830803"/>
    <application state="true" name="ss_xml_parser" vsn="3.14.0.70231.929374"
crc="6b53c933f0014cd9ab51cabd3a5ad0f116c231a2" description="Xml parser for SS definitions."
from="/usr/lib/ecss/ecss-pa-sip/lib/ss_xml_parser-3.14.0.70231.929374"/>
    <application state="true" name="ssh" vsn="4.7.6"
crc="1eefb3f7ce7ffffb428e6caaa7cd9b47802997a8" description="SSH-2 for Erlang/OTP" from="/usr/lib/
erlang/lib/ssh-4.7.6"/>
    <application state="true" name="compiler" vsn="7.3.2"
crc="2f072ba0f52e4d009cb7c8eb955a222bfc241fa5" description="ERTS CXC 138 10" from="/usr/lib/
erlang/lib/compiler-7.3.2"/>
    <application state="true" name="restfs_client_lib" vsn="3.14.0.70231.735871"
crc="6bddb92d87aba530d4d486ca0ac5e067eea98a77" description="RestFS client" from="/usr/lib/ecss/
ecss-pa-sip/lib/restfs_client_lib-3.14.0.70231.735871"/>
    <application state="true" name="ecss_cron_lib" vsn="3.14.0.70231.970465"
crc="a4c0b5f468f0ada7e0fff722e60059ea0272e6431" description="ECSS cron lib " from="/usr/lib/ecss/
ecss-pa-sip/lib/ecss_cron_lib-3.14.0.70231.970465"/>
    <application state="true" name="ecss_adapter_utils" vsn="3.14.0.70231.750971"
crc="9ce160c159d6c0281e618f1fc1e9c48e93cda162" description="Utilites for protocol adapters"
from="/usr/lib/ecss/ecss-pa-sip/lib/ecss_adapter_utils-3.14.0.70231.750971"/>
    <application state="true" name="xmerl" vsn="1.3.20"
crc="5b954cf2fcd78b3d3165a1a7656297f364233255" description="XML parser" from="/usr/lib/erlang/
lib/xmerl-1.3.20"/>
    <application state="true" name="rps_control_agent" vsn="3.14.0.70231.516803"
crc="ab7cfa30aeb727e74e62b0bfe52bfb467cafa943" description="RPS Control agent" from="/usr/lib/
ecss/ecss-pa-sip/lib/rps_control_agent-3.14.0.70231.516803"/>
    <application state="true" name="gen_socket" vsn="0.1"
crc="3303d4abcef51ed873695d308672d838eb310b8d" description="generic socket toolkit" from="/usr/
lib/ecss/ecss-pa-sip/lib/gen_socket-0.1"/>
    <application state="true" name="inets" vsn="7.0.7"
crc="167ddddd3c71e2f6ebfcbbebb2c69ca213ca7c" description="INETC CXC 138 49" from="/usr/lib/
erlang/lib/inets-7.0.7"/>
    <application state="true" name="eltex_stdlib" vsn="3.14.0.849.975066"
crc="772568d3b542e81069c4648fe5469159c2fa2cbb" description="Library with common modules from
Eltex" from="/usr/lib/ecss/ecss-pa-sip/lib/eltex_stdlib-3.14.0.849.975066"/>

```

```

<application state="true" name="ecss_lock_manager_lib" vsn="3.14.0.70231.037637"
crc="9a564e0e551527e31dac016dd785148af00881c6" description="Overcluster resource lock manager
over AMQP" from="/usr/lib/ecss/ecss-pa-sip/lib/ecss_lock_manager_lib-3.14.0.70231.037637"/>
  <application state="true" name="rps_statistics_agent" vsn="3.14.0.70231.936329"
crc="5a0e22e2791cf11610045bda5af08ad9c37a6aa3" description="RPS Statistics agents" from="/usr/
lib/ecss/ecss-pa-sip/lib/rps_statistics_agent-3.14.0.70231.936329"/>
    <application state="true" name="xml_common_lib" vsn="3.14.0.70231.053231"
crc="eb25a58885118009d15e4b906633178cfa99dd9c" description="Xml common library" from="/usr/lib/
ecss/ecss-pa-sip/lib/xml_common_lib-3.14.0.70231.053231"/>
      <application state="true" name="oasys" vsn="3.14.0.70231.753438"
crc="642673f54ad4d990ee475cfae583dd86c04a8df" description="OASys Core App" from="/usr/lib/ecss/
ecss-pa-sip/lib/oasys-3.14.0.70231.753438"/>
        <application state="true" name="ssw_ranges" vsn="3.14.0.70231.905144"
crc="4b210d0ec8975fbb7fb9d302ae3484b401a8c77" description="SSW range routines" from="/usr/lib/
ecss/ecss-pa-sip/lib/ssw_ranges-3.14.0.70231.905144"/>
          <application state="true" name="rm_agent" vsn="3.14.0.70231.831815"
crc="1071c5451f809239cf45abed305aefdba3092f59" description="RM Agent - agent for remote access to
rm_lib" from="/usr/lib/ecss/ecss-pa-sip/lib/rm_agent-3.14.0.70231.831815"/>
            <application state="true" name="qsig_asn_lib" vsn="3.14.0.70231.318655"
crc="882f3405ea1903498f00e8868b01caa168825e77" description="QSIG asn1 Lib" from="/usr/lib/ecss/
ecss-pa-sip/lib/qsig_asn_lib-3.14.0.70231.318655"/>
              <application state="true" name="qsig_coder" vsn="3.14.0.70231.101476"
crc="815e755231d3456ab59d1b222ab53ce60f7ad074" description="Realisation of QSIG encoder/decoder"
from="/usr/lib/ecss/ecss-pa-sip/lib/qsig_coder-3.14.0.70231.101476"/>
                <application state="true" name="ssw_helper_lib" vsn="3.14.0.70231.405314"
crc="3653de571876597faa9c947b1712558911fb9078" description="Routines for internal SSW usage"
from="/usr/lib/ecss/ecss-pa-sip/lib/ssw_helper_lib-3.14.0.70231.405314"/>
                  <application state="true" name="stdlib" vsn="3.8.2"
crc="c3ef8ecae59da05055d8122945c4c9b2acef4a46" description="ERTS CXC 138 10" from="/usr/lib/
erlang/lib/stdlib-3.8.2"/>
                    <application state="true" name="ecss_shell" vsn="3.14.0.70231.393543"
crc="86c4d4b047578e959b473ea24504205f1a571d69" description="ECSS Shell" from="/usr/lib/ecss/ecss-
pa-sip/lib/ecss_shell-3.14.0.70231.393543"/>
                      <application state="true" name="tring" vsn="3.14.0.70231.820830"
crc="b52243b7171d83e22dd0c90a1bb04b98b0af2890" description="Distributed supervising system, based
on amqp 0.10 protocol" from="/usr/lib/ecss/ecss-pa-sip/lib/tring-3.14.0.70231.820830"/>
                        <application state="true" name="oasys_ccn_service" vsn="3.14.0.70231.956065"
crc="5d9cc3c6fa3b43c4e4e261a2a4c790dc70fba2b" description="OASys Cocon Service" from="/usr/lib/
ecss/ecss-pa-sip/lib/oasys_ccn_service-3.14.0.70231.956065"/>
                          <application state="true" name="mycelium_ccnt_service" vsn="3.14.0.70231.259585"
crc="66d916c1052f02f5cc3b15b9a715128726c7961d" description="AMQP10 cocon service and commands"
from="/usr/lib/ecss/ecss-pa-sip/lib/mycelium_ccnt_service-3.14.0.70231.259585"/>
                            <application state="true" name="ecss_cpt_server" vsn="3.14.0.70231.191629"
crc="9f090499b7abbb05ebadfdfeb0ddbca46c71d1f4" description="Call-process tracer server" from="/
usr/lib/ecss/ecss-pa-sip/lib/ecss_cpt_server-3.14.0.70231.191629"/>
                                <application state="true" name="rps_alarm_agent" vsn="3.14.0.70231.752162"
crc="31058463f1c81ad07e141f93d677f3d8a5f5f7c1" description="RPS Alarm agent" from="/usr/lib/ecss/
ecss-pa-sip/lib/rps_alarm_agent-3.14.0.70231.752162"/>

```



```

<application state="true" name="ssl" vsn="9.2.3.1"
crc="d401ea98d54f58bb523b91db10feb68997a218fa" description="Erlang/OTP SSL application" from="/
usr/lib/erlang/lib/ssl-9.2.3.1"/>
<application state="true" name="syntax_tools" vsn="2.1.7"
crc="e04c6deada607b0ff95f5bc369fcb954aa0d8141" description="Syntax tools" from="/usr/lib/erlang/
lib/syntax_tools-2.1.7"/>
<application state="true" name="ecss_tring" vsn="3.14.0.70231.889696"
crc="19f5fd4eaae736ad9aaa4b762639daf6f116023a" description="ECSS wrap for distributed supervising
system "Tring"" from="/usr/lib/ecss/ecss-pa-sip/lib/ecss_tring-3.14.0.70231.889696"/>
<application state="true" name="eradius" vsn="3.14.0.51.515289"
crc="34bc54423cc2d676d711fb541e0af4c7514a11ae" description="RADIUS authentication/accounting
library" from="/usr/lib/ecss/ecss-pa-sip/lib/eradius-3.14.0.51.515289"/>
<application state="true" name="rps" vsn="3.14.0.70231.621393"
crc="59e49dca7dd44eb705f91a17c4e08225ae9f9fba" description="Report Processing System" from="/usr/
lib/ecss/ecss-pa-sip/lib/rps-3.14.0.70231.621393"/>
<application state="true" name="rps_log_agent" vsn="3.14.0.70231.156018"
crc="aa4ddafe1947b96080ed8c688c7f18be738cbfbf" description="RPS Log agent" from="/usr/lib/ecss/
ecss-pa-sip/lib/rps_log_agent-3.14.0.70231.156018"/>
<application state="true" name="mycelium" vsn="3.14.0.431.877455"
crc="cd407eb10ab2ec62bb4a8e5d13e504f740d8cf3a" description="Mycelium common lib" from="/usr/lib/
ecss/ecss-pa-sip/lib/mycelium-3.14.0.431.877455"/>
<application state="true" name="ecss_env" vsn="3.14.0.70231.242509"
crc="eeb8206531cca0380378344888f7085ed7c9f689" description="ECSS envermoment support lib" from="/
usr/lib/ecss/ecss-pa-sip/lib/ecss_env-3.14.0.70231.242509"/>
<application state="true" name="rm_lib" vsn="3.14.0.70231.831685"
crc="26705794ce5d3718505c48e65870fe347675a6eb" description="Perfomance Routing Library" from="/
usr/lib/ecss/ecss-pa-sip/lib/rm_lib-3.14.0.70231.831685"/>
<application state="true" name="obj_data" vsn="3.14.0.70231.549986"
crc="ea9f6c606f2dd70716e36e90f8cb2d9df2e1ef15" description="OASys Object Data Manipulations"
from="/usr/lib/ecss/ecss-pa-sip/lib/obj_data-3.14.0.70231.549986"/>
<application state="true" name="ecss_cpt_client" vsn="3.14.0.70231.954622"
crc="12114e8d1caaea06d5b9da26fcc1f856117c081b" description="Call-process tracer client" from="/
usr/lib/ecss/ecss-pa-sip/lib/ecss_cpt_client-3.14.0.70231.954622"/>
<application state="true" name="ecss_mlpp_options_lib" vsn="3.14.0.70231.397761"
crc="6c1577da30f47cd03114ba0b8f328786bfddf688" description="ECSS-10 MLPP options" from="/usr/lib/
ecss/ecss-pa-sip/lib/ecss_mlpp_options_lib-3.14.0.70231.397761"/>
<application state="true" name="log_server_ccn_service" vsn="3.14.0.70231.734967"
crc="80b5d416de1155fb279810145548d0a614ddf573" description="Log Server CCN Service" from="/usr/
lib/ecss/ecss-pa-sip/lib/log_server_ccn_service-3.14.0.70231.734967"/>
<application state="true" name="lpm_common" vsn="3.14.0.70231.871866"
crc="08b258f3d70da9472a7756b7760c86b0e2be346f" description="LPM common functions" from="/usr/lib/
ecss/ecss-pa-sip/lib/lpm_common-3.14.0.70231.871866"/>
<application state="true" name="public_key" vsn="1.6.6"
crc="a5af343b683bf63e05327dbd7df5d5d79913251f" description="Public key infrastructure" from="/
usr/lib/erlang/lib/public_key-1.6.6"/>
<application state="true" name="ecss_ssh" vsn="3.14.0.70231.264069"
crc="bbca9e46b9f0466680d61b40734898422e5da2cb" description="ECSS SSH terminals sup. lib" from="/
usr/lib/ecss/ecss-pa-sip/lib/ecss_ssh-3.14.0.70231.264069"/>

```



```

    <application state="true" name="lager" vsn="3.4.3-9-g5219ae2"
crc="846305929eac5dee6864d6ab7d48f25c903131c1" description="Empty Logging Lager2" from="/usr/lib/
ecss/ecss-pa-sip/lib/lager-3.4.3-9-g5219ae2"/>
    <application state="true" name="recon" vsn="2.3.6"
crc="e284571289227922e47d1db12ef9112dcdf1fc58" description="Diagnostic tools for production use"
from="/usr/lib/ecss/ecss-pa-sip/lib/recon-2.3.6"/>
    <application state="true" name="cmn_tools" vsn="3.14.0.70231.070567"
crc="8f1d998def0fe5c0775e06d76711b3be06e835d7" description="Common usefull tools" from="/usr/lib/
ecss/ecss-pa-sip/lib/cmn_tools-3.14.0.70231.070567"/>
    <application state="true" name="hep" vsn="0.1.0"
crc="b1bee5e0ae0f2c1adfc0bf12b02ed3cddb9adcc" description="hep - Homer Encapsulation Protocol"
from="/usr/lib/ecss/ecss-pa-sip/lib/hep-0.1.0"/>
    <application state="true" name="jsx" vsn="1.4.5"
crc="6e27d6a517e1f81b72ea7952683d546bf551398e" description="a streaming, evented json parsing
toolkit" from="/usr/lib/ecss/ecss-pa-sip/lib/jsx-1.4.5"/>
    <application state="true" name="statistics_lib" vsn="3.14.0.70231.714734"
crc="0e849f0a6bb0c98e149e51b71506b58b35d04dad" description="Q752 Statistics lib" from="/usr/lib/
ecss/ecss-pa-sip/lib/statistics_lib-3.14.0.70231.714734"/>
    <application state="true" name="ecss_conference_lib" vsn="3.14.0.70231.196644"
crc="231ed8093ed34c81523d042766463952014418b1" description="ECSS Conference lib" from="/usr/lib/
ecss/ecss-pa-sip/lib/ecss_conference_lib-3.14.0.70231.196644"/>
    <application state="true" name="pt_lib" vsn="3.14.0.33.179948"
crc="1d6f52e93f66524bf52be74ebbbe69c53ad73ec" description="Parse transform library" from="/usr/
lib/ecss/ecss-pa-sip/lib/pt_lib-3.14.0.33.179948"/>
    <application state="true" name="cocon" vsn="3.14.0.70231.970649"
crc="9d464cb284814fd94515dd380376184c777e2719" description="COMmands and CONsoles" from="/usr/
lib/ecss/ecss-pa-sip/lib/cocon-3.14.0.70231.970649"/>
    <application state="true" name="rtopman" vsn="3.14.0.70231.110741"
crc="e1c323cde2293aaba5d30ffac511b5de5c654232" description="Resourse TOPology MANagement Library"
from="/usr/lib/ecss/ecss-pa-sip/lib/rtopman-3.14.0.70231.110741"/>
    <application state="true" name="esqlite" vsn="0.4.0"
crc="6c94565e9a926606150cf6522aceb0c34b14ae77" description="sqlite nif interface" from="/usr/lib/
ecss/ecss-pa-sip/lib/esqlite-0.4.0"/>
    <application state="true" name="sip_support" vsn="3.14.0.70231.643643"
crc="f0ab61f0322f680af224a43694e8162ebed439c1" description="Support of SIP RFC extensions"
from="/usr/lib/ecss/ecss-pa-sip/lib/sip_support-3.14.0.70231.643643"/>
    <application state="true" name="sip_ssw_intercom" vsn="3.14.0.70231.408462"
crc="5c547334d7fd4940b1a49132c10e158786544448" description="SIP(I/T) protocol adapter of ECSS-10"
from="/usr/lib/ecss/ecss-pa-sip/lib/sip_ssw_intercom-3.14.0.70231.408462"/>
    <application state="true" name="necron" vsn="3.14.0.205.647555"
crc="2464d1f04d7ab9887a9c282c3331b4a4c75c6e51" description="Scheduler of tasks (cron like)"
from="/usr/lib/ecss/ecss-pa-sip/lib/necron-3.14.0.205.647555"/>
    <application state="true" name="sas1" vsn="3.3"
crc="d92bd4401c950591932f347cb66f48731df3bcb7" description="SASL CXC 138 11" from="/usr/lib/
erlang/lib/sas1-3.3"/>
    <application state="true" name="os_mon" vsn="2.4.7"
crc="d040120c316144715dae024d47c8c226f1992c83" description="CPO CXC 138 46" from="/usr/lib/
erlang/lib/os_mon-2.4.7"/>

```

```

    <application state="true" name="meck" vsn="0.8.12"
crc="58165be1810a8b9ec2cf10cb194a1631582202cc" description="A mocking framework for Erlang"
from="/usr/lib/ecss/ecss-pa-sip/lib/meck-0.8.12"/>
    <application state="true" name="ecss_cpt_lib" vsn="3.14.0.70231.141004"
crc="b7921c81d9c90312fecce0c45eb87702cad7be20" description="Call-process tracer library." from="/
usr/lib/ecss/ecss-pa-sip/lib/ecss_cpt_lib-3.14.0.70231.141004"/>
    <application state="true" name="exmpp" vsn="3.14.0.1357.450563"
crc="94b3443686cdeb0b9586b582da1485be3520e757" description="XMPP/Jabber protocol oriented XML
library" from="/usr/lib/ecss/ecss-pa-sip/lib/exmpp-3.14.0.1357.450563"/>
    <application state="true" name="ion" vsn="3.14.0.179.740603"
crc="5cd66c4d371c9a96785d87f77473ca9d9e9e84fc" description="ION pretty printer" from="/usr/lib/
ecss/ecss-pa-sip/lib/ion-3.14.0.179.740603"/>
    <application state="true" name="ecss_system_options_lib" vsn="3.14.0.70231.513460"
crc="f2f6f1880aaec67f8f908d6a7f3f23827d4c6419" description="ECSS-10 system options" from="/usr/
lib/ecss/ecss-pa-sip/lib/ecss_system_options_lib-3.14.0.70231.513460"/>
    <application state="true" name="rtopdds" vsn="3.14.0.70231.033977"
crc="dfe4dc437e25bd12252866fd68d003effa98ea1c" description="RTOP Domain Directory Service"
from="/usr/lib/ecss/ecss-pa-sip/lib/rtopdds-3.14.0.70231.033977"/>
    <application state="true" name="tools" vsn="3.1.0.1"
crc="1a0d4f44bb0376e3ccb4b9e1e6ebb3f6b09702d9" description="DEVT00LS CXC 138 16" from="/usr/lib/
erlang/lib/tools-3.1.0.1"/>
    <application state="true" name="pa_acp_io" vsn="3.14.0.70231.147885"
crc="852c7c0782c2f6d93c127f2fecf520725a1af85c" description="ACP transport application" from="/
usr/lib/ecss/ecss-pa-sip/lib/pa_acp_io-3.14.0.70231.147885"/>
    <application state="true" name="asn1" vsn="5.0.8"
crc="0beb75f03ca05225fe727725c02f2525103f50e9" description="The Erlang ASN1 compiler version
5.0.8" from="/usr/lib/erlang/lib/asn1-5.0.8"/>
    <application state="true" name="rps_event_agent" vsn="3.14.0.70231.715850"
crc="210a9424dd23a0ec7bcb5d321209fccc53dfaa5f" description="RPS Event agent" from="/usr/lib/ecss/
ecss-pa-sip/lib/rps_event_agent-3.14.0.70231.715850"/>
    <application state="true" name="runtime_tools" vsn="1.13.2"
crc="9fe7385499bf63f057ec47fe229682d1c8aefc2d" description="RUNTIME_TOOLS" from="/usr/lib/erlang/
lib/runtime_tools-1.13.2"/>
    <application state="true" name="ecss_mgmt" vsn="3.14.0.70231.994292"
crc="d5d09f4c8cf418846f870164bd9ccfb7eff1bbb2" description="ECSS Management system" from="/usr/
lib/ecss/ecss-pa-sip/lib/ecss_mgmt-3.14.0.70231.994292"/>
    <application state="true" name="lpm_server" vsn="3.14.0.70231.957898"
crc="b1ff83470b37f9ebf9e1b8fd84df6a1c66ee0a3a" description="Licence Property Manager" from="/usr/
lib/ecss/ecss-pa-sip/lib/lpm_server-3.14.0.70231.957898"/>
    <application state="true" name="mycelium_client" vsn="3.14.0.907.657726"
crc="7a1faa6d5efc312be87b0cf6c6a9cef5b5b319ef" description="Mycelium AMQP 10 erlang client"
from="/usr/lib/ecss/ecss-pa-sip/lib/mycelium_client-3.14.0.907.657726"/>
    <application state="true" name="ssw_util_lib" vsn="3.14.0.70231.111335"
crc="a93ea6564aca46ae5f3d787fed445a28fb3a6408" description="Routines for internal SSW usage"
from="/usr/lib/ecss/ecss-pa-sip/lib/ssw_util_lib-3.14.0.70231.111335"/>
    <application state="true" name="rtopdds_ccn_service" vsn="3.14.0.70231.315690"
crc="e1343510acd497731b07c47d5d764f71b2760996" description="RTOP DDS COCON Service" from="/usr/
lib/ecss/ecss-pa-sip/lib/rtopdds_ccn_service-3.14.0.70231.315690"/>

```

```
<application state="true" name="rtop_agent" vsn="3.14.0.70231.867109"
crc="03a0de3dc5ec59a9c6b0c44cef2d5d6c936ee38b" description="RTOP AGENT support lib" from="/usr/
lib/ecss/ecss-pa-sip/lib/rtop_agent-3.14.0.70231.867109"/>
  <application state="true" name="rps_common_lib" vsn="3.14.0.70231.588032"
crc="6c8e16548082cc043029671eb9663cb2f246f726" description="RPS Common library" from="/usr/lib/
ecss/ecss-pa-sip/lib/rps_common_lib-3.14.0.70231.588032"/>
    <application state="true" name="acp_lib" vsn="3.14.0.70231.111284"
crc="c5112504420cc3581e3be6629e8b4b92376e715f" description="ACP library" from="/usr/lib/ecss/
ecss-pa-sip/lib/acp_lib-3.14.0.70231.111284"/>
  </out>
```

Нс_node_releases_info

Команда возвращает информацию о релизах ноды.

Аналог команды в CoCon:

node/<NODE>/info releases

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/node_releases_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="releaseType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="version" type="xs:string" use="required"/>
    <xs:attribute name="build_crc" type="xs:string" use="required"/>
    <xs:attribute name="release_crc" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="node" type="nodeType" minOccurs="0" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="release" type="releaseType" minOccurs="0" maxOccurs="unbounded"
/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

Запрос: http://192.168.1.21:9999/commands/node_releases_info

```

<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_releases_info.xsd">
  <node name="core1@ecss1" />
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_relea
ses_info.xsd">
  <release name="ecss-core" node="core1@ecss1" version="3.14.4.327" build_crc="f5df6c336a2d883c
5a78065e79e5976a983fad44" release_crc="da39a3ee5e6b4b0d3255bfef95601890afd80709"/>
</out>
```

Пример 2:Запрос: http://192.168.1.21:9999/commands/node_releases_info

```
<?xml version="1.0" encoding="UTF-8" ?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_relea
ses_info.xsd">
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_relea
ses_info.xsd">
  <release name="ecss-pa-sip" node="sip1@ecss1" version="3.14.4.327" build_crc="7d5956cc31fa5d6
72a906735c2735a1fe3495b96" release_crc="da39a3ee5e6b4b0d3255bfef95601890afd80709"/>
  <release name="ecss-mycelium" node="mycelium1@ecss1" version="3.14.4.327" build_crc="221cc33d
8cce6a4b43186c4db233a889efc53f1" release_crc="da39a3ee5e6b4b0d3255bfef95601890afd80709"/>
  <release name="ecss-pa-megaco" node="megaco1@ecss1" version="3.14.4.327" build_crc="ea55d7ea0
d6ae7a0e5881f721aeb24ae9ab6b515" release_crc="da39a3ee5e6b4b0d3255bfef95601890afd80709"/>
  <release name="ecss-mediator" node="md1@ecss1" version="3.14.4.327" build_crc="5ba46940e6ab99
123c6f8351c323f8fd350b2974" release_crc="da39a3ee5e6b4b0d3255bfef95601890afd80709"/>
  <release name="ecss-ds" node="ds1@ecss1" version="3.14.4.327" build_crc="44e4b98eee04c237b6b1
ab7cb2e06acb3bf8282c" release_crc="da39a3ee5e6b4b0d3255bfef95601890afd80709"/>
  <release name="ecss-core" node="core1@ecss1" version="3.14.4.327" build_crc="f5df6c336a2d883c
5a78065e79e5976a983fad44" release_crc="da39a3ee5e6b4b0d3255bfef95601890afd80709"/>
</out>
```

Нс_node_statistics_info

Команда позволяет вывести статистику работы ноды.

Аналог команды в CoCon:**/node/<NODE>/info statistics****Метод HTTP-запроса:**

POST

Шаблон HTTP-запроса:

http://host:port/commands/hc_node_statistics_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="ioType">
    <xs:attribute name="input" type="xs:integer" use="required"/>
    <xs:attribute name="output" type="xs:integer" use="required"/>
  </xs:complexType>

  <xs:complexType name="garbageType">
    <xs:attribute name="times" type="xs:integer" use="required"/>
    <xs:attribute name="words" type="xs:integer" use="required"/>
  </xs:complexType>

  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:integer" use="required"/>
  </xs:complexType>

  <xs:complexType name="statisticOutType">
    <xs:all>
      <xs:element name="context_switches" type="valueType" />
      <xs:element name="reductions" type="valueType" />
      <xs:element name="garbage_collection" type="garbageType" />
      <xs:element name="io" type="ioType"/>
      <xs:element name="run_queue" type="valueType" />
      <xs:element name="runtime" type="valueType" />
      <xs:element name="wall_clock" type="valueType" />
    </xs:all>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="node" type="nodeType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="statistics" type="statisticOutType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/node_statistics_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_statistics_info.xsd">
  <node name="core1@ecss1" />
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_statistics_info.xsd">
  <statistics>
    <context_switches value="9217921"/>
    <reductions value="4094816063"/>
    <garbage_collection times="1836547" words="7151131890"/>
    <io input="59375617" output="498987640"/>
    <run_queue value="0"/>
    <runtime value="582174"/>
    <wall_clock value="26672912"/>
  </statistics>
</out>
```

hc_node_system_info

Команда позволяет вывести системную статистику работы ноды.

Аналог команды в CoCon:

node/<NODE>/info system

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/hc_node_system_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:


```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd" />

  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="integerType">
    <xs:attribute name="value" type="xs:integer" use="required"/>
  </xs:complexType>

  <xs:complexType name="booleanType">
    <xs:attribute name="value" type="xs:boolean" use="required"/>
  </xs:complexType>

  <xs:complexType name="stringType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="total_usedType">
    <xs:attribute name="total" type="xs:integer" use="required"/>
    <xs:attribute name="used" type="xs:integer" use="optional"/>
  </xs:complexType>

  <xs:complexType name="allocated_areasType" >
    <xs:sequence>
      <xs:element name="sys_misc" type="total_usedType" />
      <xs:element name="static" type="total_usedType" />
      <xs:element name="atom_space" type="total_usedType" />
      <xs:element name="atom_table" type="total_usedType" />
      <xs:element name="module_table" type="total_usedType" />
      <xs:element name="export_table" type="total_usedType" />
      <xs:element name="export_list" type="total_usedType" />
      <xs:element name="register_table" type="total_usedType" />
      <xs:element name="fun_table" type="total_usedType" />
      <xs:element name="module_refs" type="total_usedType" />
      <xs:element name="loaded_code" type="total_usedType" />
      <xs:element name="dist_table" type="total_usedType" />
      <xs:element name="node_table" type="total_usedType" />
      <xs:element name="bits_bufs_size" type="total_usedType" />
      <xs:element name="bif_timer" type="total_usedType" />
      <xs:element name="link_lh" type="total_usedType" />
      <xs:element name="process_table" type="total_usedType" />
      <xs:element name="port_table" type="total_usedType" />
      <xs:element name="ets_misc" type="total_usedType" />
    </xs:sequence>
  </xs:complexType>

```

```

<xs:complexType name="check_ioListType">
  <xs:sequence>
    <xs:element name="check_io" type="check_ioType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="check_ioType">
  <xs:sequence>
    <xs:element name="name" type="stringType" />
    <xs:element name="primary" type="stringType" />
    <xs:element name="fallback" type="stringType" />
    <xs:element name="kernel_poll" type="stringType" />
    <xs:element name="memory_size" type="integerType" />
    <xs:element name="total_poll_set_size" type="integerType" />
    <xs:element name="fallback_poll_set_size" type="integerType" />
    <xs:element name="lazy_updates" type="booleanType" />
    <xs:element name="pending_updates" type="integerType" />
    <xs:element name="batch_updates" type="booleanType" />
    <xs:element name="concurrent_updates" type="booleanType" />
    <xs:element name="max_fds" type="integerType" />
  </xs:sequence>
</xs:complexType>

<xs:complexType name="garbage_collectionType">
  <xs:attribute name="max_heap_size" type="xs:integer" use="optional"/>
  <xs:attribute name="min_bin_vheap_size" type="xs:integer" use="optional"/>
  <xs:attribute name="min_heap_size" type="xs:integer" use="optional"/>
  <xs:attribute name="fullsweep_after" type="xs:integer" use="optional"/>
</xs:complexType>

<xs:complexType name="systemOutType">
  <xs:all>
    <xs:element name="allocated_areas" type="allocated_areasType" />
    <xs:element name="c_compiler_used" type="stringType" />
    <xs:element name="check_ios" type="check_ioListType" />
    <xs:element name="compat_rel" type="integerType"/>
    <xs:element name="cpu_topology" type="stringType" />
    <xs:element name="creation" type="integerType" />
    <xs:element name="debug_compiled" type="booleanType" />
    <xs:element name="driver_version" type="stringType" />
    <xs:element name="elib_malloc" type="booleanType" />
    <xs:element name="fullsweep_after" type="integerType" />
    <xs:element name="garbage_collection" type="garbage_collectionType" />
    <xs:element name="heap_type" type="stringType" />
    <xs:element name="kernel_poll" type="booleanType" />
    <xs:element name="logical_processors" type="integerType" />
    <xs:element name="machine" type="stringType" />
    <xs:element name="min_heap_size" type="integerType" />
    <xs:element name="min_bin_vheap_size" type="integerType" />
    <xs:element name="modified_timing_level" type="stringType" />
  </xs:all>
</xs:complexType>

```

```

<xs:element name="multi_scheduling" type="stringType" />
<xs:element name="multi_scheduling_blockers" type="stringType" />
<xs:element name="otp_release" type="stringType" />
<xs:element name="process_count" type="integerType" />
<xs:element name="process_limit" type="integerType" />
<xs:element name="scheduler_bind_type" type="stringType" />
<xs:element name="scheduler_bindings" type="stringType" />
<xs:element name="scheduler_id" type="integerType" />
<xs:element name="schedulers" type="integerType" />
<xs:element name="schedulers_online" type="integerType" />
<xs:element name="smp_support" type="booleanType" />
<xs:element name="system_version" type="stringType" />
<xs:element name="system_architecture" type="stringType" />
<xs:element name="threads" type="booleanType" />
<xs:element name="thread_pool_size" type="integerType" />
<xs:element name="trace_control_word" type="integerType" />
<xs:element name="version" type="stringType" />
<xs:element name="wordsize" type="integerType" />
<xs:element name="ets_limit" type="integerType" minOccurs="0" />
</xs:all>
</xs:complexType>

<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="node" type="nodeType" minOccurs="1" maxOccurs="1"/>
    </xs:all>
  </xs:complexType>
</xs:element>

<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="system" type="systemOutType" minOccurs="0" maxOccurs="unbounded" />
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="1" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Запрос: http://192.168.1.21:9999/commands/node_system_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_system_info.xsd">
  <node name="ds1@ecss1"/>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_syste
m_info.xsd">
  <system>
    <allocated_areas>
      <sys_misc total="191168"/>
      <static total="527312"/>
      <atom_space total="1179936" used="1152205"/>
      <atom_table total="935921"/>
      <module_table total="1140984"/>
      <export_table total="1526940"/>
      <export_list total="6655616"/>
      <register_table total="1692"/>
      <fun_table total="77258"/>
      <module_refs total="136272"/>
      <loaded_code total="42647139"/>
      <dist_table total="2083"/>
      <node_table total="443"/>
      <bits_bufs_size total="0"/>
      <bif_timer total="0"/>
      <process_table total="3145728"/>
      <port_table total="786432"/>
      <ets_misc total="2097152"/>
    </allocated_areas>
```

```

    <allocator value="{glibc,[2,27],
[sys_alloc,temp_alloc,sl_alloc,std_alloc,ll_alloc,eheap_alloc,
ets_alloc,fix_alloc,literal_alloc,exec_alloc,binary_alloc,
driver_alloc,mseg_alloc,sys_aligned_alloc,literal_mmap], [{sys_alloc,[{e,true},{m,libc},{tt,
131072},{tp,0}]], {temp_alloc,[{e,true}, {t,true}, {atags,false}, {ramv,false}, {sbct,524288},
{asbcst,4145152}, {rsbcst,90}, {rsbcmt,80}, {rmbcmt,100}, {mmbcs,131072}, {mmbc,
18446744073709551615}, {mmsbc,256}, {lmbcs,10485760}, {smbcs,1048576}, {mbcgs,10}, {acul,0},
{acnl,0}, {acfml,0}, {mbsd,3}, {as,gf}]], {sl_alloc,[{e,true}, {t,true}, {atags,false},
{ramv,false}, {sbct,524288}, {asbcst,4145152}, {rsbcst,80}, {rsbcmt,80}, {rmbcmt,50}, {mmbcs,
65536}, {mmbc,18446744073709551615}, {mmsbc,256}, {lmbcs,5242880}, {smbcs,524288}, {mbcgs,10},
{acul,0}, {acnl,0}, {acfml,0}, {as,aoffcbf}]], {std_alloc,[{e,true}, {t,true}, {atags,false},
{ramv,false}, {sbct,524288}, {asbcst,4145152}, {rsbcst,20}, {rsbcmt,80}, {rmbcmt,50}, {mmbcs,
65536}, {mmbc,18446744073709551615}, {mmsbc,256}, {lmbcs,5242880}, {smbcs,524288}, {mbcgs,10},
{acul,0}, {acnl,0}, {acfml,0}, {as,aoffcbf}]], {ll_alloc,[{e,true}, {t,true}, {atags,false},
{ramv,false}, {sbct,18446744073709551615}, {asbcst,0}, {rsbcst,0}, {rsbcmt,0}, {rmbcmt,0},
{mmbcs,1048576}, {mmbc,18446744073709551615}, {mmsbc,0}, {lmbcs,5242880}, {smbcs,524288},
{mbcgs,10}, {acul,0}, {acnl,0}, {acfml,0}, {as,aoffcbf}]], {eheap_alloc,[{e,true}, {t,true},
{atags,false}, {ramv,false}, {sbct,524288}, {asbcst,4145152}, {rsbcst,50}, {rsbcmt,80}, {rmbcmt,
50}, {mmbcs,262144}, {mmbc,18446744073709551615}, {mmsbc,256}, {lmbcs,5242880}, {smbcs,524288},
{mbcgs,10}, {acul,0}, {acnl,0}, {acfml,0}, {as,aoffcbf}]], {ets_alloc,[{e,true}, {t,true},
{atags,false}, {ramv,false}, {sbct,524288}, {asbcst,4145152}, {rsbcst,20}, {rsbcmt,80}, {rmbcmt,
50}, {mmbcs,65536}, {mmbc,18446744073709551615}, {mmsbc,256}, {lmbcs,5242880}, {smbcs,524288},
{mbcgs,10}, {acul,0}, {acnl,0}, {acfml,0}, {as,aoffcbf}]], {fix_alloc,[{e,true}, {t,true},
{atags,false}, {ramv,false}, {sbct,524288}, {asbcst,4145152}, {rsbcst,20}, {rsbcmt,80}, {rmbcmt,
50}, {mmbcs,65536}, {mmbc,18446744073709551615}, {mmsbc,256}, {lmbcs,5242880}, {smbcs,524288},
{mbcgs,10}, {acul,0}, {acnl,0}, {acfml,0}, {as,aoffcbf}]], {literal_alloc,[{e,true}, {t,false},
{atags,false}, {ramv,false}, {sbct,18446744073709551615}, {asbcst,0}, {rsbcst,0}, {rsbcmt,0},
{rmbcmt,0}, {mmbcs,1048576}, {mmbc,18446744073709551615}, {mmsbc,0}, {lmbcs,10485760}, {smbcs,
1048576}, {mbcgs,10}, {acul,0}, {acnl,0}, {acfml,0}, {as,aobf}]], {exec_alloc,[{e,true},
{t,false}, {atags,false}, {ramv,false}, {sbct,18446744073709551615}, {asbcst,0}, {rsbcst,0},
{rsbcmt,0}, {rmbcmt,0}, {mmbcs,0}, {mmbc,18446744073709551615}, {mmsbc,0}, {lmbcs,10485760},
{smbcs,1048576}, {mbcgs,10}, {acul,0}, {acnl,0}, {acfml,0}, {as,aobf}]], {binary_alloc,[{e,true},
{t,true}, {atags,true}, {ramv,false}, {sbct,524288}, {asbcst,4145152}, {rsbcst,20}, {rsbcmt,80},
{rmbcmt,50}, {mmbcs,65536}, {mmbc,18446744073709551615}, {mmsbc,256}, {lmbcs,5242880}, {smbcs,
524288}, {mbcgs,10}, {acul,0}, {acnl,0}, {acfml,0}, {as,aoffcbf}]], {driver_alloc,[{e,true},
{t,true}, {atags,true}, {ramv,false}, {sbct,524288}, {asbcst,4145152}, {rsbcst,20}, {rsbcmt,80},
{rmbcmt,50}, {mmbcs,65536}, {mmbc,18446744073709551615}, {mmsbc,256}, {lmbcs,5242880}, {smbcs,
524288}, {mbcgs,10}, {acul,0}, {acnl,0}, {acfml,0}, {as,aoffcbf}]], {test_alloc,[{e,false}]],
{mseg_alloc,[{amcbf,4194304},{rmbcf,20},{mcs,10}]}, {alloc_util,[{mmc,18446744073709551615},{ycs,
1048576},{sac,true}]], {erts_mmap,[{scs,0}]}, {instr,[{t,false}]}, {lock_physical_memory,no}]]"/>
    <build_type value="opt"/>
    <c_compiler_used value="{gnuc,{7,4,0}}"/>
    <check_ios>
      <check_io>
        <name value="erts_poll"/>
        <primary value="poll"/>
        <kernel_poll value="false"/>
        <memory_size value="25240"/>
        <total_poll_set_size value="1"/>
        <lazy_updates value="true"/>
      </check_io>
    </check_ios>
  </pre>

```

```

    <pending_updates value="0"/>
    <batch_updates value="false"/>
    <concurrent_updates value="false"/>
    <fallback value="true"/>
    <max_fds value="1024"/>
    <active_fds value="0"/>
    <poll_threads value="0"/>
  </check_io>
  <check_io>
    <name value="erts_poll"/>
    <primary value="epoll"/>
    <kernel_poll value="epoll"/>
    <memory_size value="56"/>
    <total_poll_set_size value="16"/>
    <lazy_updates value="false"/>
    <pending_updates value="0"/>
    <batch_updates value="false"/>
    <concurrent_updates value="true"/>
    <fallback value="false"/>
    <max_fds value="1024"/>
    <active_fds value="0"/>
    <poll_threads value="0"/>
  </check_io>
  <check_io>
    <name value="erts_poll"/>
    <primary value="epoll"/>
    <kernel_poll value="epoll"/>
    <memory_size value="57400"/>
    <total_poll_set_size value="23"/>
    <lazy_updates value="false"/>
    <pending_updates value="0"/>
    <batch_updates value="false"/>
    <concurrent_updates value="true"/>
    <fallback value="false"/>
    <max_fds value="1024"/>
    <active_fds value="0"/>
    <poll_threads value="1"/>
  </check_io>
</check_ios>
<compat_rel value="21"/>
<cpu_topology value="[processor, [{core, {logical, 0}}, {core, {logical, 1}}]]"/>
<creation value="1"/>
<debug_compiled value="false"/>
<driver_version value="3.3"/>
<elib_malloc value="false"/>
<ets_limit value="100000"/>
<fullsweep_after value="65535"/>
<garbage_collection max_heap_size="0" min_bin_vheap_size="46422" min_heap_size="233"
fullsweep_after="65535"/>
<heap_type value="private"/>

```

```

<kernel_poll value="true"/>
<logical_processors value="2"/>
<machine value="BEAM"/>
<min_heap_size value="233"/>
<min_bin_vheap_size value="46422"/>
<modified_timing_level value="undefined"/>
<multi_scheduling value="enabled"/>
<multi_scheduling_blockers value="[]"/>
<otp_release value="21"/>
<port_parallelism value="false"/>
<port_count value="29"/>
<port_limit value="65536"/>
<process_count value="628"/>
<process_limit value="262144"/>
<scheduler_bind_type value="unbound"/>
<scheduler_bindings value="{unbound,unbound}"/>
<scheduler_id value="1"/>
<schedulers value="2"/>
<schedulers_online value="2"/>
<smp_support value="true"/>
<system_version value="Erlang/OTP 21 [erts-10.3.5.1] [source] [64-bit] [smp:2:2] [ds:
2:2:10] [async-threads:8] [hipe] "/>
<system_architecture value="x86_64-pc-linux-gnu"/>
<threads value="true"/>
<thread_pool_size value="8"/>
<trace_control_word value="0"/>
<version value="10.3.5.1"/>
<wordsize value="8"/>
</system>
</out>

```

Команда позволяет просмотреть информацию о сервисах ноды.

Аналог команды в CoCon:

`/node/<NODE>/service`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/hc_node_service_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="service" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:simpleType name = "stateType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="started"/>
      <xs:enumeration value="stopped"/>
      <xs:enumeration value="crashed"/>
      <xs:enumeration value="unkown"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="serviceType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
    <xs:attribute name="state" type="stateType" use="required"/>
  </xs:complexType>

  <xs:complexType name="statusType">
    <xs:attribute name="service" type="xs:string" use="required"/>
    <xs:attribute name="current" type="stateType" use="required"/>
    <xs:attribute name="info" type="xs:string" use="required"/>
    <xs:attribute name="old" type="stateType" use="optional"/>
    <xs:attribute name="changed" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="node" type="nodeType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="service" type="serviceType" minOccurs="1" maxOccurs="unbounded" /
>
        <xs:element name="status" type="statusType" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```


Пример:Запрос: http://192.168.1.21:9999/commands/node_service_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_service_info.xsd">
  <node name="core1@ecss1" />
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_service_info.xsd">
  <service name="ecss-cm" description="ECSS Configuration Manager" state="started"/>
  <service name="dds" description="DDS management service" state="started"/>
  <service name="rps" description="RPS management service" state="started"/>
  <service name="ecss-core" description="ECSS Core subsystem" state="started"/>
  <service name="ecss-node-manager" description="ECSS MGMT Node Manager Service" state="started"
  />
  <service name="log" description="Log Server Management Service" state="started"/>
  <service name="notifier" description="Notifier service" state="started"/>
  <service name="tring" description="ECSS Tring Service" state="started"/>
  <service name="modificator-manager" description="Modification/Adaptation Manager Service"
  state="started"/>
  <service name="amqp" description="AMQP10 Management Service" state="started"/>
</out>
```

Запрос: http://192.168.1.21:9999/commands/node_service_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_service_info.xsd">
  <node name="core1@ecss1" service="amqp" />
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_service_info.xsd">
  <status service="amqp" current="started" old="stopped" changed="2020/02/13 00:31:37" info=""/>
</out>
```

Нс_node_service

Команда позволяет управлять сервисами ноды.

Аналог команды в CoCon:

`/node/<NODE>/service`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/node_service`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name = "commandType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="run"/>
      <xs:enumeration value="stop"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="nodeType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="service" type="xs:string" use="required"/>
    <xs:attribute name="command" type="commandType" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="node" type="nodeType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/node_service

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="node_service.xsd">
  <node command="stop" name="ds1@ecss1" service="log" />
</in>

```

Ответ: 201

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="node_service.xsd">
  <node command="run" name="ds1@ecss2" service="log" />
</in>

```

Ответ: 201

Нс_node_show_cert

Команда позволяет управлять сервисами ноды.

Аналог команды в CoCon:

/node/<NODE>/show-cert

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:http://host:port/commands/node_show_cert**Код HTTP-ответа:**

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd" />
  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="node" type="xs:string" use="required"/>
            <xs:attribute name="cert_name" type="xs:string" use="required"/>
            <xs:attribute name="text" type="xs:string"/>
            <xs:attribute name="noout" type="xs:boolean"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="cert" type="xs:base64Binary" minOccurs="0"/>
        <xs:element name="error" type="errorType" minOccurs="0"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/node_show_cert

- [Hc_notifier_options_set](#)

[Hc_notifier_options_clean](#)

Команда установки параметров по умолчанию для параметров службы уведомлений.

Аналог команды в CoCon:

`/cluster/mediator/<CLUSTER>/notifier/clean`

`/domain/<DOMAIN>/notifier/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/notifier_options_clean`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean properties of notifier service.
      Usage: http://server:9999/commands/notifier_options_clean.
      Implementation: hc_notifier_options_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/notifier_options_clean

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_notifier_options_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="jabber_port"/>
        <option name="email_port"/>
        <option name="email"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_notifier_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="email"/>
        <success option="email_port"/>
        <success option="jabber_port"/>
      </domain>
    </domain_options>
  </result>
</out>

```

[Hc_notifier_options_info](#)

Команда просмотра информации о службе уведомлений.

Аналог команды в CoCon:

`/cluster/mediator/<CLUSTER>/notifier/info`

`/domain/<DOMAIN>/notifier/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/notifier_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of notifier service.
      Usage: http://server:9999/commands/notifier\_options\_info.
      Implementation: hc_notifier_options_info.erl.
      Git: git@git.eltex.loc:8080/httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- Types -->
  <xs:simpleType name="jabberPortType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="1024"/>
      <xs:maxInclusive value="65535"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="resendCntMaxType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="10000"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="bufferLimitType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="1000"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="reconnectTimerType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="1000000"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="jabberVersionType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="0.0"/>
      <xs:enumeration value="1.0"/>
    </xs:restriction>
  </xs:simpleType>

```

```

    </xs:restriction>
</xs:simpleType>

<xs:complexType name="NotifierOptionsType">
  <xs:complexContent>
    <xs:extension base="baseOptionsType">
      <xs:attribute name="jid" type="xs:string" use="optional"/>
      <xs:attribute name="jabber_password" type="xs:string" use="optional">
        <xs:annotation>
          <xs:documentation xml:lang="en">
            In case of strong security restrictions password may be shown as *****
          </xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="jabber_port" type="jabberPortType" use="optional"/>
      <xs:attribute name="jabber_enable" type="booleanType" use="optional"/>
      <xs:attribute name="jabber_buffer_limit" type="bufferLimitType" use="optional"/>
      <xs:attribute name="jabber_reconnect_timer" type="reconnectTimerType" use="optional"/>
      <xs:attribute name="jabber_version" type="jabberVersionType" use="optional"/>
      <xs:attribute name="email" type="xs:string" use="optional"/>
      <xs:attribute name="email_server" type="xs:string" use="optional"/>
      <xs:attribute name="email_port" type="xs:string" use="optional"/>
      <xs:attribute name="email_username" type="xs:string" use="optional"/>
      <xs:attribute name="email_password" type="xs:string" use="optional">
        <xs:annotation>
          <xs:documentation xml:lang="en">
            In case of strong security restrictions password may be shown as *****
          </xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="email_enable" type="booleanType" use="optional"/>
      <xs:attribute name="email_buffer_limit" type="bufferLimitType" use="optional"/>
      <xs:attribute name="email_resend_timer" type="reconnectTimerType" use="optional"/>
      <xs:attribute name="email_protection_connection" type="xs:string" use="optional"/>
      <xs:attribute name="email_resend_count_max" type="resendCntMaxType" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>

```

```

    <xs:all>
      <xs:element name="domain_options" type="domainInfoInType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/notifier_options_info

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_notifier_options_info.xsd">
<request>
  <domain_options domain="biysk.local"/>
</request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_notifier_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="NotifierOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="NotifierOptionsType"
          jid="ecss-10@jabber.eltex.loc"
          jabber_version="0.0" jabber_reconnect_timer="10"
          jabber_port="5222" jabber_password="*****"
          jabber_enable="false" jabber_buffer_limit="50"
          email_username="ecss10biysk@mail.ru"
          email_server="smtp.mail.ru" email_resend_timer="30"
          email_resend_count_max="5"
          email_protection_connection="ssl_tls"
          email_port="465" email_password="*****"
          email_enable="false" email_buffer_limit="50"
          email="ecss10biysk@mail.ru"/>
      </domain>
    </domain_options>
  </result>
</out>

```

hc_notifier_options_set

Команда настройки параметров службы уведомлений.

Аналог команды в CoCon:

/cluster/mediator/<CLUSTER>/notifier/set

/domain/<DOMAIN>/notifier/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/notifier_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set notifier service properties
      Usage: http://server:9999/commands/notifier\_options\_set.
      Implementation: hc_notifier_options_set.erl.
      Git: git@git.eltex.loc:8080.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- Types -->
  <xs:simpleType name="jabberPortType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="1024"/>
      <xs:maxInclusive value="65535"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="resendCntMaxType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="10000"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="bufferLimitType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="1000"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="reconnectTimerType">
    <xs:restriction base="integerRangeType">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="1000000"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="jabberVersionType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="0.0"/>
      <xs:enumeration value="1.0"/>
    </xs:restriction>
  </xs:simpleType>

```

```

</xs:simpleType>

<xs:complexType name="NotifierOptionsType">
  <xs:complexContent>
    <xs:extension base="baseOptionsType">
      <xs:attribute name="jid" type="xs:string" use="optional"/>
      <xs:attribute name="jabber_password" type="xs:string" use="optional"/>
      <xs:attribute name="jabber_port" type="jabberPortType" use="optional"/>
      <xs:attribute name="jabber_enable" type="booleanType" use="optional"/>
      <xs:attribute name="jabber_buffer_limit" type="bufferLimitType" use="optional"/>
      <xs:attribute name="jabber_reconnect_timer" type="reconnectTimerType" use="optional"/>
      <xs:attribute name="jabber_version" type="jabberVersionType" use="optional"/>
      <xs:attribute name="email" type="xs:string" use="optional"/>
      <xs:attribute name="email_server" type="xs:string" use="optional"/>
      <xs:attribute name="email_port" type="xs:string" use="optional"/>
      <xs:attribute name="email_username" type="xs:string" use="optional"/>
      <xs:attribute name="email_password" type="xs:string" use="optional"/>
      <xs:attribute name="email_enable" type="booleanType" use="optional"/>
      <xs:attribute name="email_buffer_limit" type="bufferLimitType" use="optional"/>
      <xs:attribute name="email_resend_timer" type="reconnectTimerType" use="optional"/>
      <xs:attribute name="email_protection_connection" type="xs:string" use="optional"/>
      <xs:attribute name="email_resend_count_max" type="resendCntMaxType" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<!-- resultType -->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

```

```

        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/notifier_options_set

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_notifier_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="NotifierOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="NotifierOptionsType" email_protection_connection="starttls"
email_buffer_limit="50" jabber_buffer_limit="0" jabber_enable="true" email_enable="false"
email_port="default" email_resend_count_max="5"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Код ответа:

200

Ответ:


```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_notifier_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="email_buffer_limit"/>
        <success option="email_enable"/>
        <success option="email_port"/>
        <success option="email_protection_connection"/>
        <success option="email_resend_count_max"/>
        <success option="jabber_buffer_limit"/>
        <success option="jabber_enable"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_notifier_send_test - Управление отправкой нотификаций через службу уведомлений

Нс_notifier_send_test_msg

Команда позволяет отправить тестовое сообщение через службу уведомлений на электронную почту, Jabber.

Аналог команды в CoCon:

/domain/<DOMAIN>/notifier/send_test_msg

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/notifier_send_test_msg

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема xml файла с запросом/ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />
  <xs:simpleType name="typeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="email"/>
      <xs:enumeration value="jabber"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="type" type="typeType" use="required"/>
            <xs:attribute name="address" type="xs:string" use="required"/>
            <xs:attribute name="msg" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!--Positive case-->
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <!--Or error-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/notifier_send_test_msg

```
<?xml version="1.0"?>
  <in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_notifier_send_test.xsd">
    <request domain="biysk.local" type="email" address="asz@sibnet.ru" msg="test http message"/>
  </in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_notifier_send_test.xsd">
  <response>
    <ok/>
  </response>
</out>
```

Нс_np_numbers - Управление списками номеров плана нумерации

- [Нс_np_add_number](#)
- [Нс_np_delete_number2](#)
- [Нс_np_rename_number](#)
- [Нс_np_list_number](#)
- [Нс_np_bind_number](#)
- [Нс_np_unbind_number](#)

Нс_np_add_number

Команда позволяет добавить номер в план нумерации виртуальной АТС.

Аналог команды в CoCon:

domain/**<DOMAIN>**/**np/numbers/add**

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/np_add_number

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <!-- <error> -->
  <xs:complexType name="npErrorType">
    <xs:attribute name="cmd" type="xs:token" use="required" />
    <xs:attribute name="reason" type="xs:string" use="required" />
    <xs:attribute name="body" type="xs:string" use="optional" />
    <!-- entity="[nonempty_string]" -->
    <xs:attribute name="entity" type="xs:normalizedString" use="optional" />
  </xs:complexType>
  <!-- </error> -->

  <!-- Error reasons description -->
  <xs:simpleType name="npErrorReasonType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="not_exists" />
      <xs:enumeration value="unknown_error" />
      <!-- More reasons can be added here -->
    </xs:restriction>
  </xs:simpleType>
  <!-- End of error reasons description -->

  <!-- Error reasons description -->
  <xs:simpleType name="npNumberResultType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="ok" />
      <xs:enumeration value="already_added" />
      <!-- More reasons can be added here -->
    </xs:restriction>
  </xs:simpleType>
  <!-- End of error reasons description -->

  <xs:complexType name="npResult">
    <xs:sequence>
      <xs:element name="number" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string"/>
          <xs:attribute name="result" type="npNumberResultType"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="npSuccess">
    <xs:sequence>
      <xs:element name="np" type="npResult" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

```

```

</xs:complexType>

<xs:complexType name="npErrors">
  <xs:sequence>
    <xs:element name="np" minOccurs="0" maxOccurs="unbounded">
      <xs:complexType>
        <xs:attribute name="name" type="xs:string"/>
        <xs:attribute name="reason" type="npErrorReasonType"/>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="npNumberingPlan">
  <xs:sequence>
    <xs:element name="number" minOccurs="0" maxOccurs="unbounded">
      <xs:complexType>
        <xs:attribute name="name" type="xs:string" use="required"/>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>

<xs:complexType name="npList">
  <xs:sequence>
    <xs:element name="np" type="npNumberingPlan" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="np_list" type="npList"/>
          </xs:sequence>
          <xs:attribute name="domain" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>

<!-- out-->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>

```

```

<!--Positive case-->
<xs:element name="response">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="success" type="npSuccess"/>
      <xs:element name="errors" type="npErrors"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!--Or error-->
<xs:element name="error" type="npErrorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/np_add_number

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_np_add_number.xsd">
  <request domain="biysk.local">
    <np_list>
      <np name="np_240">
        <number name="415800"/>
        <number name="333333"/>
      </np>
      <np name="city">
        <number name="240500"/>
        <number name="240318"/>
      </np>
    </np_list>
  </request>
</in>

```

Ответ:200

```

<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_add_number.xsd">
  <response>
    <success>
      <np name="np_240">
        <number name="415800" result="already_added"/>
        <number name="333333" result="ok"/>
      </np>
      <np name="city">
        <number name="240500" result="ok"/>
        <number name="240318" result="ok"/>
      </np>
    </success>
    <errors/>
  </response>
</out>

```

Нс_np_delete_number2

Команда позволяет удалить номер из плана нумерации виртуальной АТС.

Аналог команды в CoCon:

domain/**<DOMAIN>**/**np/numbers/delete**

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/np_delete_number2

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <!-- <error> -->
  <xs:complexType name="npErrorType">
    <xs:attribute name="cmd" type="xs:token" use="required" />
    <xs:attribute name="reason" type="xs:string" use="required" />
    <xs:attribute name="body" type="xs:string" use="optional" />
    <!-- entity="[nonempty_string]" -->
    <xs:attribute name="entity" type="xs:normalizedString" use="optional" />
  </xs:complexType>
  <!-- </error> -->

  <!-- Error reasons description -->
  <xs:simpleType name="npErrorReasonType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="not_exists" />
      <xs:enumeration value="unknown_error" />
      <!-- More reasons can be added here -->
    </xs:restriction>
  </xs:simpleType>
  <!-- End of error reasons description -->

  <!-- Error reasons description -->
  <xs:simpleType name="npNumberResultType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="ok" />
      <xs:enumeration value="not_exists" />
      <!-- More reasons can be added here -->
    </xs:restriction>
  </xs:simpleType>
  <!-- End of error reasons description -->

  <xs:complexType name="npResult">
    <xs:sequence>
      <xs:element name="number" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string"/>
          <xs:attribute name="result" type="npNumberResultType"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="npSuccess">
    <xs:sequence>
      <xs:element name="np" type="npResult" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

```



```

</xs:complexType>

<xs:complexType name="npErrors">
  <xs:sequence>
    <xs:element name="np" minOccurs="0" maxOccurs="unbounded">
      <xs:complexType>
        <xs:attribute name="name" type="xs:string"/>
        <xs:attribute name="reason" type="npErrorReasonType"/>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="npNumberingPlan">
  <xs:sequence>
    <xs:element name="number" minOccurs="0" maxOccurs="unbounded">
      <xs:complexType>
        <xs:attribute name="name" type="xs:string" use="required"/>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>

<xs:complexType name="npList">
  <xs:sequence>
    <xs:element name="np" type="npNumberingPlan" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="np_list" type="npList"/>
          </xs:sequence>
          <xs:attribute name="domain" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>

<!-- out-->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>

```

```

<!--Positive case-->
<xs:element name="response">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="success" type="npSuccess"/>
      <xs:element name="errors" type="npErrors"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!--Or error-->
<xs:element name="error" type="npErrorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/np_delete_number2

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_np_delete_number2.xsd">
  <request domain="biysk.local">
    <np_list>
      <np name="np_240">
        <number name="333333"/>
        <number name="444000"/>
      </np>
      <np name="city">
        <number name="240318"/>
        <number name="240500"/>
      </np>
    </np_list>
  </request>
</in>

```

Ответ 200:

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_delete_number2.xsd">
  <response>
    <success>
      <np name="np_240">
        <number name="333333" result="ok"/>
        <number name="444000" result="not_exists"/>
      </np>
      <np name="city">
        <number name="240318" result="ok"/>
        <number name="240500" result="ok"/>
      </np>
    </success>
    <errors/>
  </response>
</out>
```

hc_np_rename_number

Команда позволяет переименовать указанные номера в плане нумерации.

Аналог команды в CoCon:

`domain/<DOMAIN>/np/numbers/rename`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/np_rename_number`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <!-- <error> -->
  <xs:complexType name="npErrorType">
    <xs:attribute name="cmd" type="xs:token" use="required" />
    <xs:attribute name="reason" type="npErrorReasonType" use="required" />
    <xs:attribute name="body" type="xs:string" use="optional" />
    <!-- entity="[nonempty_string]" -->
    <xs:attribute name="entity" type="xs:normalizedString" use="optional" />
  </xs:complexType>
  <!-- </error> -->

  <!-- Error reasons description -->
  <xs:simpleType name="npErrorReasonType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="not_exists" />
      <xs:enumeration value="conflict_numbers" />
      <xs:enumeration value="unknown_error" />
      <!-- More reasons can be added here -->
    </xs:restriction>
  </xs:simpleType>
  <!-- End of error reasons description -->

  <xs:simpleType name="npRenameResultStrType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="ok" />
      <xs:enumeration value="not_exists" />
      <!-- More reasons can be added here -->
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="NPRenameResultType">
    <xs:attribute name="old_value" type="xs:string" use="required"/>
    <xs:attribute name="new_value" type="xs:string" use="required"/>
    <xs:attribute name="result" type="npRenameResultStrType" use="required"/>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="np_number" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="old_value" type="xs:string" use="required"/>
                  <xs:attribute name="new_value" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

        </xs:complexType>
    </xs:element>
</xs:sequence>
<xs:attribute name="domain" type="xs:string" use="required"/>
<xs:attribute name="np_name" type="xs:string" use = "required"/>
</xs:complexType>
</xs:element>
</xs:all>
</xs:complexType>
</xs:element>
<!-- out-->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <!--Positive case-->
            <xs:element name="response">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="np_number" type="NPRenameResultType" minOccurs="0"
maxOccurs="unbounded"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <!--Or error-->
            <xs:element name="error" type="npErrorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Запрос: http://192.168.1.21:9999/commands/np_rename_number

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_rename_number.xsd">
  <request domain="biysk.test" np_name="new">
    <np_number old_value="800" new_value="2800"/>
    <np_number old_value="900" new_value="2900"/>
  </request>
</in>

```

Ответ 200:

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_rename_number.xsd">
  <response>
    <np_number old_value="900" new_value="2900" result="ok"/>
    <np_number old_value="800" new_value="2800" result="ok"/>
  </response>
</out>
```

Нс_np_list_number

Команда просмотра списка номеров в плане нумерации и ресурсов, которые к ним привязаны.

Аналог команды в CoCon:

domain/<DOMAIN>/np/numbers/List

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/np_list_number

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <!-- <error> -->
  <xs:complexType name="npErrorType">
    <xs:attribute name="cmd" type="xs:token" use="required" />
    <xs:attribute name="reason" type="npErrorReasonType" use="required" />
    <xs:attribute name="body" type="xs:string" use="optional" />
    <!-- entity="[nonempty_string()]" -->
    <xs:attribute name="entity" type="xs:normalizedString" use="optional" />
  </xs:complexType>
  <!-- </error> -->

  <!-- Error reasons description -->
  <xs:simpleType name="npErrorReasonType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="not_exists" />
      <xs:enumeration value="unknown_error" />
      <!-- More reasons can be added here -->
    </xs:restriction>
  </xs:simpleType>
  <!-- End of error reasons description -->

  <xs:complexType name="NPNumberType">
    <xs:choice minOccurs="0" maxOccurs="1">
      <xs:element name="bridge">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
      <xs:element name="aliases">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="alias" minOccurs="0" maxOccurs="unbounded">
              <xs:complexType>
                <xs:attribute name="iface" type="xs:string" use="required"/>
                <xs:attribute name="name" type="xs:string" use="required"/>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:choice>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="npType">
    <xs:attribute name="name" type="xs:string"/>
  </xs:complexType>

```

```

<xs:complexType name="npList">
  <xs:sequence>
    <xs:element name="np" type="npType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="npInfo">
  <xs:sequence>
    <xs:element name="np_number" type="NPNumberType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string"/>
</xs:complexType>

<xs:complexType name="npResult">
  <xs:sequence>
    <xs:element name="np" type="npInfo" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="np_list" type="npList" minOccurs="0"/>
          </xs:sequence>
          <xs:attribute name="domain" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>

<!-- out-->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <!--Positive case-->
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="np_list" type="npResult"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <!--Or error-->
      <xs:element name="error" type="npErrorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

```



```

    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/np_list_number

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_np_list_number.xsd">
  <request domain="biysk.local"/>
</in>

```

Ответ:

```

<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_np_list_number.xsd">
  <response>
    <np_list>
      <np name="city">
        <np_number name="240318"/>
        <np_number name="240500"/>
      </np>
      <np name="np_240">
        <np_number name="333333"/>
        <np_number name="415800"/>
      </np>
    </np_list>
  </response>
</out>

```

Нс_np_bind_number

Команда привязки номера плана нумерации с локальным номером или исходящем бриджем.

Аналог команды в CoCon:

domain/<DOMAIN>/np/numbers/bind

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/np_bind_number

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <!-- <error> -->
  <xs:complexType name="npErrorType">
    <xs:attribute name="cmd" type="xs:token" use="required" />
    <xs:attribute name="reason" type="xs:string" use="required" />
    <xs:attribute name="body" type="xs:string" use="optional" />
    <!-- entity="[nonempty_string()]" -->
    <xs:attribute name="entity" type="xs:normalizedString" use="optional" />
  </xs:complexType>
  <!-- </error> -->

  <!--Алиасы: -->
  <!-- alias_not_exists - нет такого алиаса-->
  <!-- np_number_not_exists - нет такого номера в плане нумерации-->
  <!-- np_not_exists - нет такого плана нумерации-->
  <!-- unsupported_np_type - с данным номером связан мост-->
  <!-- np_number_already_binded - данный алиас уже привязан к этому номеру в плане нумерации-->
  <!-- np_already_binded - данный алиас уже привязан к другому номеру в плане нумерации-->
  <!--Мосты: -->
  <!-- np_number_not_exists - нет такого номера в плане нумерации-->
  <!-- np_is_busy - к данному плану нумерации уже привязан мост или алиасы-->
  <!-- np_a_not_exists - план нумерации А не существует-->
  <!-- np_b_not_exists - план нумерации Б не существует-->
  <!-- bridge_not_found - нет такого моста-->
  <!-- invalid_bridge - указанный мост настроен неправильно для использования в данном плане
нумерации-->

  <xs:simpleType name="npNumberResultType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="ok" />
      <xs:enumeration value="alias_not_exists" />
      <xs:enumeration value="np_number_not_exists" />
      <xs:enumeration value="np_not_exists" />
      <xs:enumeration value="unsupported_np_type" />
      <xs:enumeration value="np_number_already_binded" />
      <xs:enumeration value="np_already_binded" />
      <xs:enumeration value="np_is_busy" />
      <xs:enumeration value="np_a_not_exists" />
      <xs:enumeration value="np_b_not_exists" />
      <xs:enumeration value="bridge_not_found" />
      <xs:enumeration value="invalid_bridge" />
      <xs:enumeration value="unknown_error" />
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="npNumberingPlan">
    <xs:sequence>

```

```

<xs:element name="np_number" minOccurs="0" maxOccurs="unbounded">
  <xs:complexType>
    <xs:choice>
      <xs:element name="alias">
        <xs:complexType>
          <xs:attribute name="iface" type="xs:string" use="required"/>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="mode" use="required">
            <xs:simpleType>
              <xs:restriction base="xs:token">
                <xs:enumeration value="master"/>
                <xs:enumeration value="passive"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
      <xs:element name="bridge">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:choice>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>

<xs:complexType name="npList">
  <xs:sequence>
    <xs:element name="np" type="npNumberingPlan" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="npResult">
  <xs:sequence>
    <xs:element name="np_number" minOccurs="0" maxOccurs="unbounded">
      <xs:complexType>
        <xs:attribute name="name" type="xs:string"/>
        <xs:attribute name="entity" type="xs:string"/>
        <xs:attribute name="result" type="npNumberResultType"/>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string"/>
</xs:complexType>

<xs:complexType name="npResultList">

```

```

<xs:sequence>
  <xs:element name="np" type="npResult" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="np_list" type="npList"/>
          </xs:sequence>
          <xs:attribute name="domain" type="xs:string"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>

<!-- out-->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <!--Positive case-->
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="np_list" type="npResultList"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <!--Or error-->
      <xs:element name="error" type="npErrorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/np_bind_number

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_bind_number.xsd">
  <request domain="biysk.local">
    <np_list>
      <np name="np_240">
        <np_number name="700">
          <alias iface="061dd777ba450f12" name="240466" mode="master"/>
        </np_number>
        <np_number name="700">
          <alias iface="061dd777c14e201d" name="240467" mode="passive"/>
        </np_number>
        <np_number name="800">
          <bridge name="biysk.local_to_biysk.test"/>
        </np_number>
      </np>
    </np_list>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_bind_number.xsd">
  <response>
    <np_list>
      <np name="np_240">
        <np_number name="700" entity="240466" result="ok"/>
        <np_number name="700" entity="240467" result="ok"/>
        <np_number name="800" entity="biysk.local_to_biysk.test" result="ok"/>
      </np>
    </np_list>
  </response>
</out>

```

Нс_np_unbind_number

Команда отменяет привязку номера/номеров плана нумерации с локальным номером или исходящем бриджем.

Аналог команды в CoCon:

domain/<DOMAIN>/np/numbers/unbind

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/np_unbind_number

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- <error> -->
  <xs:complexType name="npErrorType">
    <xs:attribute name="cmd" type="xs:token" use="required" />
    <xs:attribute name="reason" type="xs:string" use="required" />
    <xs:attribute name="body" type="xs:string" use="optional" />
    <!-- entity="[nonempty_string()]" -->
    <xs:attribute name="entity" type="xs:normalizedString" use="optional" />
  </xs:complexType>
  <!-- </error> -->
  <!--Алиасы: -->
  <!-- np_not_exists - нет такого плана нумерации -->
  <!-- np_number_not_binded - нет такого номера в плане нумерации, либо он не привязан к
внутреннему номеру -->
  <!--Бриджи: -->
  <!-- bridge_not_found - нет такого бриджа -->
  <!-- np_a_not_exists - нет такого плана нумерации -->
  <!-- np_not_busy - номер не привязан к бриджу -->
  <!-- np_number_not_exists - нет такого номера в плане нумерации -->
  <!-- invalid_bridge - внутренняя ошибка бриджа -->
  <xs:simpleType name="npNumberResultType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="ok" />
      <xs:enumeration value="np_not_exists" />
      <xs:enumeration value="np_number_not_binded" />
      <xs:enumeration value="bridge_not_found" />
      <xs:enumeration value="np_a_not_exists" />
      <xs:enumeration value="np_not_busy" />
      <xs:enumeration value="np_number_not_exists" />
      <xs:enumeration value="invalid_bridge" />
      <xs:enumeration value="unknown_error" />
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="npNumberingPlan">
    <xs:sequence>
      <xs:element name="np_number" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:choice>
            <!-- Выполнить unbind для алиаса -->
            <xs:element name="alias">
              <xs:complexType>
                <xs:attribute name="iface" type="xs:string" use="required"/>
                <xs:attribute name="name" type="xs:string" use="required"/>
              </xs:complexType>
            </xs:element>
            <!-- Выполнить unbind для бриджа -->
            <xs:element name="bridge">

```



```

        <xs:complexType>
            <!-- Если имя бриджа неизвестно, можно указать в качестве имени
бриджа any -->
            <xs:attribute name="name" type="xs:string" use="required"/>
        </xs:complexType>
    </xs:element>
    <!-- Выполнить unbind автоматически в зависимости от того к чему привязан
номер плана нумерации -->
    <xs:element name="clean"/>
</xs:choice>
    <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>
</xs:element>
</xs:sequence>
    <xs:attribute name="name" type="xs:string" use = "required"/>
</xs:complexType>
<xs:complexType name="npList">
    <xs:sequence>
        <xs:element name="np" type="npNumberingPlan" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="npResult">
    <xs:sequence>
        <xs:element name="np_number" minOccurs="0" maxOccurs="unbounded">
            <xs:complexType>
                <xs:attribute name="name" type="xs:string"/>
                <xs:attribute name="result" type="npNumberResultType"/>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string"/>
</xs:complexType>
<xs:complexType name="npResultList">
    <xs:sequence>
        <xs:element name="np" type="npResult" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<!-- in -->
<xs:element name="in">
    <xs:complexType>
        <xs:all>
            <xs:element name="request">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="np_list" type="npList"/>
                    </xs:sequence>
                    <xs:attribute name="domain" type="xs:string"/>
                </xs:complexType>
            </xs:element>
        </xs:all>
    </xs:complexType>

```

```

    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <!--Positive case-->
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="np_list" type="npResultList"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <!--Or error-->
      <xs:element name="error" type="npErrorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/np_unbind_number

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_unbind_number.xsd">
  <request domain="biysk.local">
    <np_list>
      <np name="np_240">
        <np_number name="700">
          <alias iface="061dd777ba450f12" name="240466"/>
        </np_number>
      </np>
    </np_list>
  </request>
</in>

```

Ответ:200

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_unbind_number.xsd">
  <response>
    <np_list>
      <np name="np_240">
        <np_number name="700" result="ok"/>
      </np>
    </np_list>
  </response>
</out>
```

Нс_np - Управление настройками плана нумерации

- [Нс_np_declare](#)
- [Нс_np_info](#)
- [Нс_np_set](#)
- [Нс_np_clean](#)
- [Нс_np_remove](#)
- [Нс_np_list](#)
- [Нс_np_db_list](#)

Нс_np_declare

Команда позволяет создать новый план нумерации.

Аналог команды в CoCon:

domain/<DOMAIN>/np/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/np_declare

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="domainType">
    <xs:sequence>
      <xs:element name="numbering_plan" type="npType" minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="npType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="digitmap" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="1"
maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!-- Positive case -->
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

        <!--Or error-->
        <xs:element name="error" type="errorType"/>
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/np_declare

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_declare.xsd">
  <request>
    <domain name="test.biysk">
      <numbering_plan name="new" digitmap="xxxxxx" description="test_subscribers"/>
    </domain>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_declare.xsd">
  <response>
    <ok/>
  </response>
</out>

```

Ответ:200

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_declare.xsd">
  <error cmd="hc_np_declare" reason="already_declared" body="{error,already_exists},<<"new">>
  }]" />
</out>
```

Нс_np_info

Команда позволяет просмотреть список и свойства планов нумерации.

Аналог команды в CoCon:

`domain/<DOMAIN>/np/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/np_info`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="numPlanType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="digitmap" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>
  <xs:complexType name="domainOutType">
    <xs:sequence>
      <xs:element name="numbering_plan" type="numPlanType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="1"
maxOccurs="1"/>
              <xs:element name="numbering_plan" type="domainType" minOccurs="0"
maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!-- Positive case -->
        <xs:element name="response">

```

```

        <xs:complexType>
            <xs:all>
                <xs:element name="domain" type="domainOutType"/>
            </xs:all>
        </xs:complexType>
    </xs:element>
    <!--Or error-->
    <xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/np_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_info.xsd">
  <request>
    <domain name="test.biysk"/>
    <numbering_plan name="new"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_info.xsd">
  <response>
    <domain name="test.biysk">
      <numbering_plan name="new" digitmap="xxxxxx" description="test_subscribers"/>
    </domain>
  </response>
</out>

```

Нс_np_set

Команда позволяет изменить свойства плана нумерации.

Аналог команды в CoCon:

`domain/<DOMAIN>/np/set`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/np_set`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="domainType">
    <xs:sequence>
      <xs:element name="numbering_plan" type="npType" minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="npType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="digitmap" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!--Positive case-->
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>

```

```

        <xs:element name="ok"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<!--Or error-->
    <xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/np_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_set.xsd">
  <request>
    <domain name="test.biysk">
      <numbering_plan name="new" digitmap="2xxxxx" description="City subscribers">
        <property name="ni" value="local"/>
        <property name="nai" value="nationalNumber"/>
      </numbering_plan>
      <numbering_plan name="local_np" description="Local subscribers" />
    </domain>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_set.xsd">
  <response>
    <ok/>
  </response>
</out>

```

hc_np_clean

Команда позволяет удалить определенное свойство из номерного плана.

Аналог команды в CoCon:**domain/<DOMAIN>/np/clean**

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/np_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="domainType">
    <xs:sequence>
      <xs:element name="numbering_plan" type="npType" minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="npType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!--Positive case-->
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

        <!--Or error-->
        <xs:element name="error" type="errorType"/>
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/np_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_np_clean.xsd">
<request>
  <domain name="p.city">
    <numbering_plan name="city">
      <property name="ni"/>
      <property name="nai"/>
    </numbering_plan>
  </domain>
</request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_clean.xsd">
  <response>
    <ok/>
  </response>
</out>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_clean.xsd">
  <error cmd="hc_np_clean" reason="not_exists" body="{{error,not_exists},"x-plan"}"/>
</out>
```

Нс_np_remove

Команда позволяет удалить существующий номерной план

Аналог команды в CoCon:

`domain/<DOMAIN>/np/remove`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/np_remove`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1"/>
              <xs:element name="numbering_plan" type="domainType" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!-- Positive case -->
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <!-- Or error -->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/np_remove


```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_np_remove.xsd">
<request>
  <domain name="biysk.local" />
  <numbering_plan name="city" />
</request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_np_remove.xsd">
  <response>
    <ok/>
  </response>
</out>
```

Повторно отправленный запрос вернет следующий вывод:

Ответ:404

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_np_remove.xsd">
  <error cmd="hc_np_remove" reason="not_exists" body="{{error,not_exists},"city"}/>
</out>
```

Нс_np_list

Команда предназначена для отображения списка имен планов нумерации в определенном домене.

Аналог команды в CoCon:

/domain/<DOMAIN>/np/info

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/np_list

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема xml файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- NPTYPE -->
  <xs:complexType name="NPType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="np" type="NPType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/np_list

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_np_list.xsd">
  <request domain="biysk.local"/>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_np_list.xsd">
  <response>
    <np name="np_240" description="other"/>
    <np name="city" description="city_subscriber"/>
  </response>
</out>
```

Нс_np_db_list

Команда предназначена для получения из реестра плана нумерации списка городов/регионов/операторов.

Аналог команды в CoCon:

/domain/<DOMAIN>/np/db/list

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/np_db_list

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема xml файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain" type="domainType" minOccurs="1"
maxOccurs="1"/>
              <xs:element name="type_table" type="domainType" minOccurs="1"
maxOccurs="1"/>
              <xs:element name="filter" type="domainType" minOccurs="0"
maxOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!--Positive case-->
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="element" type="xs:string"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <!--Or error-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/np_db_list

```
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_np_db_list.xsd">
<request>
  <domain name="biysk.local"/>
  <type_table name="region"/>
  <filter name="НОВ"/>
</request>
</in>
```

Ответ:

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_np_db_list.xsd">
  <response>
    <element>НОВОСИБИРСКАЯ ОБЛ.</element>
    <element>ИВАНОВСКАЯ ОБЛ.</element>
    <element>НОВГОРОДСКАЯ ОБЛ.</element>
    <element>УЛЬЯНОВСКАЯ ОБЛ.</element>
  </response>
</out>
```

Нс_oct_cp_options - Управление системными настройками ecss-core

- [Нс_oct_cp_options_clean](#)
- [Нс_oct_cp_options_info](#)
- [Нс_oct_cp_options_set](#)

[Нс_oct_cp_options_clean](#)

Команда сброса системных настроек ECSS-CORE.

Аналог команды в CoCon:

domain/<DOMAIN>/core/system/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/oct_cp_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean value of requested property(es).
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"
minOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:http://192.168.1.21:9999/commands/oct_cp_options_clean

```
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="oct_cp_options_clean.xsd">
  <request>
    <domain_options>
      <system/>
      <domain name="test.ats">
        <option name="amqp_cp_acp_ttl"/>
        <option name="amqp_cp_acpr_ttl"/>
        <option name="max_hops_count"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="oct_cp_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="test.ats">
        <success option="amqp_cp_acp_ttl"/>
        <success option="amqp_cp_acpr_ttl"/>
        <success option="max_hops_count"/>
      </domain>
    </domain_options>
  </result>
</out>
```

[Hc_oct_cp_options_info](#)

Команда просмотра информации о подсистеме RADIUS авторизации (access).

Аналог команды в CoCon:**domain/<DOMAIN>/core/system/info****Метод HTTP-запроса:**

POST

Шаблон HTTP-запроса:

http://host:port/commands/oct_cp_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show value of requested property(es).
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <xs:complexType name="coreCPOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="max_hops_count" type="integerType" use="optional" />
        <xs:attribute name="amqp_cp_acpr_ttl" type="integerType" use="optional" />
        <xs:attribute name="amqp_cp_acp_ttl" type="integerRangeType" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainInfoInType" minOccurs="
1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```
</xs:element>
</xs:schema>
```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/oct_cp_options_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="oct_cp_options_info.xsd">
  <request>
    <domain_options domain="test.ats"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="oct_cp_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="coreCP0ptionsType"/>
      </system>
      <domain name="test.ats">
        <options amqp_cp_acp_ttl="30000" amqp_cp_acpr_ttl="4000" max_hops_count="5"
xs:type="coreCP0ptionsType"/>
      </domain>
    </domain_options>
  </result>
</out>
```

[Hc_oct_cp_options_set](#)

Команда изменения системных настроек ECSS-CORE.

Аналог команды в CoCon:

domain/<DOMAIN>/core/system/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/oct_cp_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set value of requested property(es).
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <xs:complexType name="coreCPOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="max_hops_count" type="integerType" use="optional" />
        <xs:attribute name="amqp_cp_acpr_ttl" type="integerType" use="optional" />
        <xs:attribute name="amqp_cp_acp_ttl" type="integerRangeType" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainSetInType"
minOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```
</xs:element>
</xs:schema>
```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/oct_cp_options_set

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="oct_cp_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="coreCPOptionsType"/>
      </system>
      <domain name="test.ats">
        <options amqp_cp_acp_ttl="30000" amqp_cp_acpr_ttl="4000" max_hops_count="5"
xs:type="coreCPOptionsType"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="oct_cp_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="test.ats">
        <success option="amqp_cp_acp_ttl"/>
        <success option="amqp_cp_acpr_ttl"/>
        <success option="max_hops_count"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Нс_queue - Управление очередями виртуального call-центра

- [Нс_queue_add_agent](#)

- [Hc_queue_clean](#)
- [Hc_queue_configuration](#)
- [Hc_queue_declare](#)
- [Hc_queue_list](#)
- [Hc_queue_remove](#)
- [Hc_queue_remove_agent](#)
- [Hc_queue_set](#)

[Hc_queue_add_agent](#)

Команда предназначена для добавления агентов в выбранную очередь call-центра.

Аналог команды в CoSop:

`domain/<DOMAIN>/cc/queue/<QUEUE>/add-agent`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/queue_add_agent

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="hc_queue_common.xsd"/>
  <xs:include schemaLocation="error.xsd"/>
  <!--errorReasonType-->
  <xs:simpleType name="warningReasonType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="already_added"/>
      <xs:enumeration value="not_exists"/>
    </xs:restriction>
  </xs:simpleType>
  <!--errorsType-->
  <xs:complexType name="warningsType">
    <xs:sequence>
      <xs:element name="warning" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="agent" type="xs:string"/>
          <xs:attribute name="reason" type="warningReasonType"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="agents" type="agentsType"/>
            </xs:sequence>
            <xs:attribute name="domain" use="required"/>
            <xs:attribute name="queue" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:choice>
              <!--all agents was added-->
              <xs:element name="ok">

```

```

        <xs:complexType/>
    </xs:element>
    <!--not very bad. command works fine but some agents not added:-->
    <xs:element name="warnings" type="warningsType"/>
</xs:choice>
</xs:complexType>
</xs:element>
<!--bad. command failed:-->
<xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/queue_add_agent

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_add_agent.xsd">
  <request domain="biysk.local" queue="qltp3">
    <agents>
      <agent name = "521"/>
      <agent name = "522"/>
      <agent name = "523"/>
      <agent name = "524"/>
      <agent name = "525"/>
    </agents>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_add_agent.xsd">
  <response>
    <warnings>
      <warning agent="525" reason="already_added"/>
    </warnings>
  </response>
</out>

```

Ответ на тот же самый запрос, если агент с именем "529" не существует в контакт-центре:


```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_add_agent.xsd">
  <response>
    <warnings>
      <warning agent="529" reason="not_exists"/>
    </warnings>
  </response>
</out>
```

hc_queue_clean

Очистить указанные свойства очереди виртуального call-центра.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/queue/<QUEUE>/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/queue_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="hc_queue_common.xsd"/>
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="propertyNameType">
    <xs:attribute name="name" type="xs:string"/>
  </xs:complexType>

  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <!--request body-->
              <xs:element name="queue">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="property" type="propertyNameType" maxOccurs="unbounded"/>
                  </xs:sequence>
                <!--queue name-->
                <xs:attribute name="name" use="required"/>
              </xs:complexType>
            </xs:element>
          <!--end-->
        </xs:sequence>
        <!--request attributes-->
        <xs:attribute name="domain" use="required"/>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name = "out">
  <xs:complexType>
    <xs:choice>
      <!--Positive case-->
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="ok"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <!--Or error-->
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

```

```

    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1

Запрос: http://192.168.1.21:9999/commands/queue_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request domain="biysk.local">
    <queue name="qltp3">
      <property name="description"/>
      <property name="max_wait_time"/>
      <property name="some\custom\property"/>
    </queue>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_clean.xsd">
  <response>
    <ok/>
  </response>
</out>

```

Пример 2

Ответ (434) на предыдущий запрос если нет такой очереди:

```

<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_clean.xsd">
  <error cmd="queue_clean" reason="nocommand" body="Command not found by path /domain/
biysk.local/cc/queue/qltp4/clean."/>
</out>

```

hc_queue_configuration

Команда позволяет выбрать свойства очереди указанного call-центра из БД.

Аналог команды в CoCon:

`domain/<DOMAIN>/cc/queue/configuration <QUEUE>`

`domain/<DOMAIN>/cc/queue/<QUEUE>/configuration`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/queue_configuration

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="hc_queue_common.xsd"/>

  <!--configurationsType-->
  <xs:complexType name="configurationsType">
    <xs:sequence>
      <xs:element name="queue" type="queueType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!--in-->
  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <!--request body-->
              <xs:element name="queue" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="name" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
            <!--end-->
            </xs:sequence>
          </xs:complexType>
          <xs:attribute name="domain" type="xs:string" use="required"/>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!--out-->
  <xs:element name = "out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="queues" type="configurationsType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/queue_configuration

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_configuration.xsd">
  <request domain="biysk.local">
    <queue name="line1"/>
    <queue name="line2"/>
    <queue name="qltp3"/>
  </request>
</in>
```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_configuration.xsd">
  <response>
    <queues>
      <queue name="line1" ringback_mode="once" distribution_mode="random"
max_wait_time="3600" description="" skill_based_distribution="false"
max_distribution_attempts="3" max_distribution_duration="10" window_duration="10"
remember_choice="none">
        <groups>
          <group name="ltp1"/>
        </groups>
        <agents>
          <agent name="509"/>
          <agent name="508"/>
          <agent name="507"/>
          <agent name="506"/>
          <agent name="505"/>
          <agent name="504"/>
          <agent name="503"/>
          <agent name="502"/>
          <agent name="501"/>
          <agent name="500"/>
        </agents>
        <properties>
          <property name="time_prediction\statistics\lower_threshold" value="10"/>
          <property name="time_prediction\statistics\upper_threshold" value="1800"/>
          <property name="time_prediction\statistics\max_count" value="100"/>
          <property name="time_prediction\based_on" value="defined_value"/>
          <property name="time_prediction\defined_value" value="60"/>
          <property name="time_prediction\min_values_in_statistics" value="10"/>
        </properties>
      </queue>
      <queue name="line2" ringback_mode="once" distribution_mode="random"
max_wait_time="3600" description="" skill_based_distribution="false"
max_distribution_attempts="3" max_distribution_duration="10" window_duration="10"
remember_choice="none">
        <groups>
          <group name="ltp2"/>
        </groups>
        <agents>
          <agent name="518"/>
          <agent name="517"/>
          <agent name="516"/>
          <agent name="515"/>
          <agent name="514"/>
          <agent name="513"/>
          <agent name="512"/>
        </agents>
      </queue>
    </queues>
  </response>
</out>

```

```

        <agent name="511"/>
        <agent name="510"/>
    </agents>
    <properties>
        <property name="time_prediction\statistics\lower_threshold" value="10"/>
        <property name="time_prediction\statistics\upper_threshold" value="1800"/>
        <property name="time_prediction\statistics\max_count" value="100"/>
        <property name="time_prediction\based_on" value="defined_value"/>
        <property name="time_prediction\defined_value" value="60"/>
        <property name="time_prediction\min_values_in_statistics" value="10"/>
    </properties>
    </queue>
    <queue name="qltp3" queue_length="6" ringback_mode="once" distribution_mode="random"
max_wait_time="1800" description="my queue" skill_based_distribution="false"
max_distribution_attempts="2" max_distribution_duration="20" window_duration="5"
remember_choice="none">
        <groups>
            <group name="group1"/>
            <group name="group2"/>
        </groups>
        <agents>
            <agent name="526"/>
            <agent name="525"/>
            <agent name="524"/>
            <agent name="523"/>
            <agent name="522"/>
            <agent name="521"/>
        </agents>
        <properties>
            <property name="time_prediction\statistics\lower_threshold" value="10"/>
            <property name="time_prediction\statistics\upper_threshold" value="1800"/>
            <property name="time_prediction\statistics\max_count" value="100"/>
            <property name="time_prediction\based_on" value="defined_value"/>
            <property name="time_prediction\defined_value" value="60"/>
            <property name="time_prediction\min_values_in_statistics" value="10"/>
            <property name="another\custom\property" value="[another,term]"/>
            <property name="some\custom\property" value="{erlang,term}"/>
        </properties>
    </queue>
</queues>
</response>
</out>

```

nc_queue_declare

Команда позволяет создать новую очередь.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/queue/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/queue_declare

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

hc_queue_common.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

<xs:include schemaLocation="common_types.xsd"/>
<!--distributionModeType-->
<xs:simpleType name = "distributionModeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="random"/>
    <xs:enumeration value="loa"/>
    <xs:enumeration value="mia"/>
    <xs:enumeration value="multicall"/>
    <xs:enumeration value="serial"/>
    <xs:enumeration value="cycled"/>
    <xs:enumeration value="progressive"/>
    <xs:enumeration value="manual"/>
  </xs:restriction>
</xs:simpleType>
<!--rememberChoiceType-->
<xs:simpleType name = "rememberChoiceType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="none"/>
    <xs:enumeration value="strict"/>
    <xs:enumeration value="non_strict"/>
  </xs:restriction>
</xs:simpleType>
<!--ringbackModeType-->
<xs:simpleType name = "ringbackMode">
  <xs:restriction base="xs:string">
    <xs:enumeration value="once"/>
    <xs:enumeration value="none"/>
  </xs:restriction>
</xs:simpleType>
<!--aliasAgentType-->
<xs:complexType name="aliasAgentType">
  <xs:attribute name="number" type="xs:string" use="required"/>
</xs:complexType>
<!--aliasAgentType-->
<xs:complexType name="ccAgentType">
  <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>
<!--groupType-->
<xs:complexType name="groupType">
  <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>
<!--agentsType-->
<xs:complexType name="agentsType">
  <xs:sequence>
    <xs:element name="alias" type="aliasAgentType" minOccurs="0" maxOccurs="unbounded"/>

```

```

    <xs:element name="agent" type="ccAgentType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!--groupsType-->
<xs:complexType name="groupsType">
  <xs:sequence>
    <xs:element name="group" type="groupType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!--customPropertiesType-->
<xs:complexType name="customPropertiesType">
  <xs:sequence>
    <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Available custom properties:
            time_prediction\statistics\lower_threshold :: Lower threshold of conversation
duration in seconds. Values less then lower threshold ignored during prediction. Default is 10
seconds.
            time_prediction\statistics\upper_threshold :: Upper threshold of conversation
duration in seconds. Values greater then upper threshold ignored during prediction. Default is 30
minutes.
            time_prediction\statistics\max_count      :: Max values stored in the statistics
count. Default is 100.
            time_prediction\based_on                  :: average | most_freq | defined_value.
What value to consider during prediction. Default is average.
            average - processed conversation duration average.
            most_freq - most frequently appeared interval of conversation processing duration.
            defined_value - manually specified value of conversation processing duration.
            time_prediction\defined_value            :: Manually defined single conversation
processing time in seconds. It used in prediction if it explicitly specified by
time_prediction\based_on or if statistic values count less then
time_prediction\min_values_in_statistics. Default is 60 seconds.
            time_prediction\min_values_in_statistics  :: Min values count in the statistics to
make prediction. If values count less then time_prediction\min_values_in_statistics then queue
will use time_prediction\defined_value as estimate value of the single conversation handling
duration. Default is 10.
        </xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
<!--queueType-->
<xs:complexType name="queueType">
  <xs:all>
    <xs:element name="agents" type="agentsType" minOccurs="0"/>
    <xs:element name="groups" type="groupsType" minOccurs="0"/>
    <xs:element name="properties" type="customPropertiesType" minOccurs="0"/>
  </xs:all>
  <xs:attribute name="name" type="xs:string" use="required"/>

```

```
<xs:attribute name="distribution_mode" type="distributionModeType" use="optional"/>
<xs:attribute name="max_wait_time" type="xs:nonNegativeInteger" use="optional"/>
<xs:attribute name="max_distribution_attempts" type="xs:nonNegativeInteger" use="optional"/>
<xs:attribute name="max_distribution_duration" type="xs:nonNegativeInteger" use="optional"/>
<xs:attribute name="ringback_mode" type="ringbackMode" use="optional"/>
<xs:attribute name="max_predicted_time" type="xs:nonNegativeInteger" use="optional"/>
<xs:attribute name="window_duration" type="xs:nonNegativeInteger" use="optional"/>
<xs:attribute name="queue_length" type="xs:nonNegativeInteger" use="optional"/>
<xs:attribute name="max_active_waiting_callbacks" type="xs:nonNegativeInteger" use="optional"/>
<xs:attribute name="description" type="xs:string" use="optional"/>
<xs:attribute name="remember_choice" type="rememberChoiceType" use="optional"/>
<xs:attribute name="skill_based_distribution" type="xs:boolean" use="optional"/>
<xs:attribute name="decline_if_no_operators" type="xs:boolean" use="optional"/>
<xs:attribute name="email" type="xs:string" use="optional"/>
<xs:attribute name="auxwork_if_no_answer" type="xs:string" use="optional"/>
<xs:attribute name="auxwork_if_reject" type="xs:string" use="optional"/>
<xs:attribute name="lock_if_no_answer" type="xs:boolean" use="optional"/>
<xs:attribute name="lock_if_reject" type="xs:boolean" use="optional"/>
<xs:attribute name="acw_status_set" type="xs:string" use="optional"/>
<xs:attribute name="callback_cooldown_timeout" type="xs:nonNegativeInteger" use="optional"/>
</xs:complexType>

</xs:schema>
```

hc_queue_declare.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="hc_queue_common.xsd"/>
  <xs:include schemaLocation="error.xsd"/>

  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="queue" type="queueType" minOccurs="1" maxOccurs="1"/>
            </xs:all>
            <xs:attribute name="domain" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence minOccurs="0" maxOccurs="1">
        <xs:element name="error" type="errorType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример 1

Запрос: http://192.168.1.21:9999/commands/queue_declare

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_declare.xsd">
  <request domain="biysk.local">
    <queue name="qltp3" distribution_mode="random" max_wait_time="1800"
    max_distribution_attempts="2" max_distribution_duration="20" window_duration="5"
    queue_length="6" description="my queue" remember_choice="none" skill_based_distribution="false">
      <agents>
        <agent name="525"/>
        <agent name="526"/>
      </agents>
      <groups>
        <group name="group1"/>
        <group name="group2"/>
      </groups>
      <properties>
        <property name="some\\custom\\property" value="{erlang, term}"/>
        <property name="another\\custom\\property" value="[another, term]"/>
      </properties>
    </queue>
  </request>
</in>

```

Ответ:

201

Пример 2

Ответ(200) на предыдущий запрос, если такая очередь уже есть в указанном виртуальном call-центре:

```

<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_declare.xsd">
  <error cmd="hc_queue_declare" reason="already_declared" body="{error, already_declared}"
  entity="qltp3"/>
</out>

```

hc_queue_list

Команда предназначена для отображения списка очередей выбранного call-центра.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/queue/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/queue_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="hc_queue_common.xsd"/>

  <xs:complexType name = "queuesType">
    <xs:sequence>
      <xs:element name = "queue" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name = "name" type="xs:string" use="required"/>
          <xs:attribute name = "groups" type="xs:string" use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:element name = "out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name = "queues" type="queuesType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/queue_list


```
<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request domain="biysk.local"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_list.xsd">
  <response>
    <queues>
      <queue name="line1" groups="ltp1"/>
      <queue name="line2" groups="ltp2"/>
      <queue name="line3" groups="ltp3"/>
      <queue name="qltp3" groups="group1,group2"/>
    </queues>
  </response>
</out>
```

Нс_queue_remove

Команда предназначена для удаления очереди из выбранного виртуального call-центра.

Аналог команды в CoСon:

domain/<DOMAIN>/cc/queue/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/queue_remove

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="hc_queue_common.xsd"/>

  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <!--request body-->
              <xs:element name="queue">
                <xs:complexType>
                  <!--queue name-->
                  <xs:attribute name="name" use="required"/>
                </xs:complexType>
              </xs:element>
              <!--end-->
            </xs:sequence>
            <xs:attribute name="domain" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name = "out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.12:9999/commands/queue_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request domain="biysk.local">
    <queue name="line3"/>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_queue_remove.xsd">
  <response>
    <ok/>
  </response>
</out>
```

hc_queue_remove_agent

Удалить агентов из указанной очереди call-центра.

Аналог команды в CoSop:

domain/<DOMAIN>/cc/queue/<QUEUE>/remove-agent

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/queue_remove_agent

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="hc_queue_common.xsd"/>
  <xs:include schemaLocation="error.xsd"/>
  <!--errorReasonType-->
  <xs:simpleType name="warningReasonType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="not_in_queue"/>
    </xs:restriction>
  </xs:simpleType>
  <!--errorsType-->
  <xs:complexType name="warningsType">
    <xs:sequence>
      <xs:element name="warning" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="agent" type="xs:string"/>
          <xs:attribute name="reason" type="warningReasonType"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <!--in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="agents" type="agentsType"/>
            </xs:sequence>
            <xs:attribute name="domain" use="required"/>
            <xs:attribute name="queue" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:choice>
              <!--all agents was removed-->
              <xs:element name="ok">
                <xs:complexType/>
              </xs:element>
            </xs:choice>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

        </xs:element>
        <!--not very bad. command works fine but some agents was not removed:-->
        <xs:element name="warnings" type="warningsType"/>
    </xs:choice>
</xs:complexType>
</xs:element>
<!--bad. command failed:-->
<xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/queue_remove_agent

```

<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request domain="biysk.local" queue="qltp3">
    <agents>
      <agent name = "524"/>
      <agent name = "525"/>
    </agents>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_remove_agent.xsd">
  <response>
    <ok/>
  </response>
</out>

```

Ответ на предыдущий запрос, если таких агентов нет в очереди:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_remove_agent.xsd">
  <response>
    <warnings>
      <warning agent="525" reason="not_in_queue"/>
      <warning agent="524" reason="not_in_queue"/>
    </warnings>
  </response>
</out>
```

HC_queue_set

Установить свойства очереди виртуального call-центра.

Аналог команды в CoCon:

domain/<DOMAIN>/cc/queue/<QUEUE>/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/queue_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="hc_queue_common.xsd"/>
  <xs:include schemaLocation="error.xsd"/>

  <xs:element name = "in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <!--request body-->
              <xs:element name="queue" type="queueType"/>
              <!--end-->
            </xs:sequence>
            <xs:attribute name="domain" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name = "out">
    <xs:complexType>
      <xs:choice>
        <!--Positive case-->
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <!--Or error-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1

Запрос:

http://192.168.1.21:9999/commands/queue_set

```
<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request domain="biysk.local">
    <queue name="qltp3" distribution_mode="mia" max_wait_time="1801"
max_distribution_attempts="3" max_distribution_duration="21" window_duration="15"
queue_length="16" description="my queue new" remember_choice="strict"
skill_based_distribution="true">
      <agents>
        <alias number="522"/>
        <agent name="524"/>
        <agent name="525"/>
      </agents>
      <groups>
        <group name="group2"/>
        <group name="group3"/>
      </groups>
      <properties>
        <property name="some\\custom\\property" value="{erlang, term1}"/>
        <property name="another\\custom\\property2" value="[another, new, term]"/>
      </properties>
    </queue>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_set.xsd">
  <response>
    <ok/>
  </response>
</out>
```

Пример 2

Ответ на предыдущий запрос если нет такой очереди:

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_queue_set.xsd">
  <error cmd="queue_set" reason="nocommand" body="Command not found by path /domain/
biysk.local/cc/queue/ltp4/set."/>
</out>
```


Hc_restfs_clusters_autocleaner - Управление настройками подсистемы Autocleaner кластера RestFS

- [Hc_restfs_clusters_autocleaner_add_rule](#)
- [Hc_restfs_clusters_autocleaner_change_rule](#)
- [Hc_restfs_clusters_autocleaner_info](#)
- [Hc_restfs_clusters_autocleaner_remove_rule](#)

[Hc_restfs_clusters_autocleaner_add_rule](#)

Команда добавляет указанное правило для подсистемы Autocleaner кластера Restfs.

Аналог команды в CoCon:

restfs/<RESTFS-CLUSTERNAME>/autocleaner/add-rule

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_clusters_autocleaners_add_rule

Код HTTP-ответа:

- 200 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="cluster" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="rule" minOccurs="1" maxOccurs="unbound
d">
                      <xs:complexType>
                        <xs:attribute name="position" type="xs:nonNegativ
eInteger" use="optional"/>
                        <xs:attribute name="file_path" type="xs:string"
use="required"/>
                        <xs:attribute name="file_mask" type="xs:string"
use="required"/>
                        <xs:attribute name="outdate" type="xs:nonNegative
Integer" use="required"/>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/restfs_clusters_autocleaner_add_rule

```
<in>
  <request>
    <cluster name="default">
      <rule position="6" file_path="test4" file_mask="*.png" outdate="10"/>
      <rule file_path="mp4" file_mask="*.mp4" outdate="7"/>
    </cluster>
  </request>
</in>
```

Ответ:

```
status: 200
content:
```

[Hc_restfs_clusters_autocleaner_change_rule](#)

Команда изменяет указанное правило для подсистемы Autocleaner кластера Restfs.

Аналог команды в CoCon:

restfs/<RESTFS-CLUSTERNAME>/autocleaner/change-rule

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_clusters_autocleaners_change_rule

Код HTTP-ответа:

- 200 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="cluster" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="rule" minOccurs="1" maxOccurs="unbounde
d">
                      <xs:complexType>
                        <xs:attribute name="position" type="xs:nonNegativ
eInteger" use="required"/>
                        <xs:attribute name="file_path" type="xs:string"
use="optional"/>
                        <xs:attribute name="file_mask" type="xs:string"
use="optional"/>
                        <xs:attribute name="outdate" type="xs:nonNegative
Integer" use="optional"/>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/restfs_clusters_autocleaner_change_rule

```
<in>
  <request>
    <cluster name="default">
      <rule position="3" file_path="test2" file_mask="*.tiff" outdate="5"/>
      <rule position="4" file_path="test5" file_mask="*.mp3" outdate="3"/>
    </cluster>
    <cluster name="mpeg">
      <rule position="1" file_path="mpeg" file_mask="*.mp4" outdate="3"/>
    </cluster>
  </request>
</in>
```

Ответ:

```
status: 201
content:
```

[Hc_restfs_clusters_autocleaner_info](#)

Команда показывает список правил подсистемы Autocleaner кластера Restfs.

Аналог команды в CoCon:

restfs/<RESTFS-CLUSTERNAME>/autocleaner/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_clusters_autocleaners_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 434 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="cluster" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="name" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="cluster" minOccurs="0" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="rule" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="position" type="xs:integer" use="required"
"/>
                  <xs:attribute name="file_path" type="xs:string" use="required"
"/>
                  <xs:attribute name="file_mask" type="xs:string" use="required"
"/>
                  <xs:attribute name="outdate" type="xs:string" use="required"/
>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="name" type="xs:string" use="required"/>
            <xs:attribute name="peer" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>

```

```

        </xs:sequence>
    </xs:complexType>
</xs:element>

</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/restfs_clusters_autocleaner_info

```

<in>
  <request>
    <cluster name="default"/>
    <cluster name="ringtones"/>
  </request>
</in>

```

Ответ:

```

status: 200
content:
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_restfs_
clusters_autocleaner_info.xsd">
  <cluster name="default" peer="http://system.restfs.ecss:9990">
    <rule position="1" file_path="mpeg" file_mask="*.mp4" outdate="3"/>
    <rule position="2" file_path="records" file_mask="*.mp3" outdate="180"/>
    <rule position="3" file_path="test2" file_mask="*.tiff" outdate="5"/>
    <rule position="4" file_path="test5" file_mask="*.mp3" outdate="3"/>
    <rule position="5" file_path="records" file_mask="*.wav" outdate="180"/>
    <rule position="6" file_path="test4" file_mask="*.png" outdate="10"/>
    <rule position="7" file_path="mp4" file_mask="*.mp4" outdate="7"/>
  </cluster>
  <cluster name="mpeg" peer="http://192.168.1.21:9990">
    <rule position="1" file_path="mpeg" file_mask="*.mp4" outdate="3"/>
    <rule position="2" file_path="records" file_mask="*.mp3" outdate="180"/>
    <rule position="3" file_path="test2" file_mask="*.tiff" outdate="5"/>
    <rule position="4" file_path="test5" file_mask="*.mp3" outdate="3"/>
    <rule position="5" file_path="records" file_mask="*.wav" outdate="180"/>
    <rule position="6" file_path="test4" file_mask="*.png" outdate="10"/>
    <rule position="7" file_path="mp4" file_mask="*.mp4" outdate="7"/>
  </cluster>
</out>

```

hc_restfs_clusters_autocleaner_remove_rule

Команда удаляет указанное правило для подсистемы Autocleaner кластера Restfs.

Аналог команды в CoCon:

`restfs/<RESTFS-CLUSTERNAME>/autocleaner/remove_rule`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/restfs_clusters_autocleaners_remove_rule`

Код HTTP-ответа:

- 200 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="cluster" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="rule" minOccurs="1" maxOccurs="unbounde
d">
                      <xs:complexType>
                        <xs:attribute name="position" type="xs:nonNegativ
eInteger" use="required"/>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/restfs_clusters_autocleaner_remove_rule

```

<in>
  <request>
    <cluster name="default">
      <rule position="3"/>
    </cluster>
    <cluster name="mpeg">
      <rule position="4"/>
    </cluster>
  </request>
</in>

```

Ответ:

```

status: 201
content:

```

Ответ в случае ошибки:

```

status: 200
content:
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_restfs_
clusters_autocleaner_remove_rule.xsd">
  <error cmd="hc_restfs_clusters_autocleaner_remove_rule" reason="high id: 4" entity="mpeg"/>
  <error cmd="hc_restfs_clusters_autocleaner_remove_rule" reason="high id: 3" entity="default"/
>
</out>

```

hc_restfs_clusters - Управление RestFS кластерами на уровне ECSS-10

- [hc_restfs_clusters_declare](#)
- [hc_restfs_clusters_list](#)
- [hc_restfs_clusters_remove](#)
- [hc_restfs_clusters_status](#)
- [hc_restfs_clusters_set](#)

hc_restfs_clusters_declare

Команда добавляет новый Restfs кластер в список кластеров Restfs.

Аналог команды в CoCon:

restfs/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:http://host:port/commands/restfs_clusters_declare**Код HTTP-ответа:**

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="cluster" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="name" type="xs:string" use="required"/>
                  <xs:attribute name="peer" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:http://192.168.1.21:9999/commands/restfs_clusters_declare

```
<in>
  <request>
    <cluster name="new_cluster1" peer="http://192.168.2.21:9990"/>
    <cluster name="new_cluster2" peer="ftp://filezilla.com:4815"/>
  </request>
</in>
```

Ответ:

```
status: 201
content:
```

[Hc_restfs_clusters_list](#)

Команда показывает список кластеров Restfs.

Аналог команды в CoCon:

restfs/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_clusters_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 434 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="cluster" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="name" type="xs:string" use="required"/>
                  <xs:attribute name="peer" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

http://192.168.1.21:9999/commands/restfs_clusters_list

```

<in>
  <request/>
</in>

```

Ответ:

```
status: 201
content:
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_restfs_
clusters_list.xsd">
  <response name="default" peer="http://system.restfs.ecss:9990" status="connected"/>
  <response name="mpeg" peer="http://192.168.1.21:9990" status="connected"/>
  <response name="new_cluster1" peer="http://192.168.2.21:9990" status="connected"/>
  <response name="new_cluster2" peer="ftp://filezilla.com:4815" status="disconnected"/>
</out>
```

[Hc_restfs_clusters_remove](#)

Команда удаляет Restfs кластер из списка кластеров Restfs.

Аналог команды в CoCon:

restfs/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_clusters_remove

Код HTTP-ответа:

- 204 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="clusterType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="cluster" type="clusterType" minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

http://192.168.1.21:9999/commands/restfs_clusters_remove

```

<in>
  <request>
    <cluster name="new_cluster2"/>
  </request>
</in>

```

Ответ успешного удаления:

```
status: 204
content:
```

Ответ удаления несуществующего кластера:

```
status: 200
content:
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_restfs_
clusters_remove.xsd">
  <error cmd="hc_restfs_clusters_remove" reason="unknown_name" entity="cluster99"/>
</out>
```

[Hc_restfs_clusters_status](#)

Команда показывает статус Restfs кластера.

Аналог команды в CoCon:

restfs/status

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_clusters_status

Код HTTP-ответа:

- 200 – в случае успеха;
- 434 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="cluster" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="name" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="cluster" minOccurs="0" maxOccurs="unbounded">
          <xs:complexType>
            <xs:attribute name="name" type="xs:string" use="optional"/>
            <xs:attribute name="error" type="xs:string" use="optional"/>
            <xs:attribute name="status" type="xs:string" use="optional"/>
            <xs:attribute name="active" type="xs:string" use="optional"/>
            <xs:attribute name="reading" type="xs:string" use="optional"/>
            <xs:attribute name="writing" type="xs:string" use="optional"/>
            <xs:attribute name="waiting" type="xs:string" use="optional"/>
            <xs:attribute name="realpath" type="xs:string" use="optional"/>
            <xs:attribute name="nginx_version" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

http://192.168.1.21:9999/commands/restfs_clusters_status

```
<in>
  <request>
    <cluster name="default"/>
    <cluster name="mpeg"/>
    <cluster name="new_cluster1"/>
  </request>
</in>
```

Ответ:

```
status: 200
content:
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_restfs_
clusters_status.xsd">
  <cluster name="default" status="Connected" realpath="/var/lib/ecss/restfs" reading="0" active="
4" waiting="2" nginx_version="1.14.0" writing="2"/>
  <cluster name="mpeg" status="Connected" realpath="/var/lib/ecss/restfs" reading="0" active="3
" waiting="1" nginx_version="1.14.0" writing="2"/>
  <cluster name="new_cluster1" status="Connected" realpath="/var/lib/ecss/restfs" reading="0"
active="2" waiting="0" nginx_version="1.14.0" writing="2"/>
</out>
```

hc_restfs_clusters_set

Команда изменяет Restfs кластер из списка кластеров Restfs'a

Аналог команды в CoCon:

restfs/change

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_clusters_set

Код HTTP-ответа:

- 204 – в случае успеха;
- 201 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="cluster" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="change" minOccurs="1" maxOccurs="1">
                      <xs:complexType>
                        <xs:attribute name="port" type="xs:positiveInteger" use="optional"/>
                        <xs:attribute name="peer" type="xs:string" use="optional"/>
                        <xs:attribute name="name" type="xs:string" use="optional"/>
                        <xs:attribute name="host" type="xs:string" use="optional"/>
                        <xs:attribute name="schema" type="xs:string" use="optional"/>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

http://192.168.1.21:9999/commands/restfs_clusters_set

```
<in>
  <request>
    <cluster name="new_cluster1">
      <change name="new_cluster4" peer="http://192.168.2.61:9990"/>
    </cluster>
  </request>
</in>
```

Ответ:

```
status: 204
content:
```

Hc_restfs_domain_autocleaner - Управление настройками подсистемы Autocleaner кластера RestFS на уровне домена

- [Hc_restfs_domain_autocleaner_add_rule](#)
- [Hc_restfs_domain_autocleaner_change_rule](#)
- [Hc_restfs_domain_autocleaner_remove_rule](#)
- [Hc_restfs_domain_autocleaner_info](#)
- [Hc_restfs_domain_autocleaner_set](#)

[Hc_restfs_domain_autocleaner_add_rule](#)

Команда предназначена для добавления определенному домену правил автоматической подчистки устаревших файлов.

Аналог команды в CoCon:

domain/<DOMAIN-NAME>/restfs/autocleaner/add-rule

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_domain_autocleaner_add_rule

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="server" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="rule" minOccurs="1" maxOccurs="unbounde
d">
                      <xs:complexType>
                        <xs:attribute name="position" type="xs:nonNegativ
eInteger" use="optional"/>
                        <xs:attribute name="file_path" type="xs:string"
use="required"/>
                        <xs:attribute name="file_mask" type="xs:string"
use="required"/>
                        <xs:attribute name="outdate" type="xs:nonNegative
Integer" use="required"/>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                  <xs:attribute name="name" type="serverType" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:simpleType name="serverType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="fax"/>
      <xs:enumeration value="call"/>
      <xs:enumeration value="sounds"/>
      <xs:enumeration value="voicemail"/>
    </xs:restriction>
  </xs:simpleType>

```

```

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример:

Запрос: http://192.168.23.34:9999/commands/restfs_domain_autocleaner_add_rule

```

<in>
  <request domain="biysk.local">
    <server name="fax" >
      <rule position="6" file_path="fax" file_mask="*.png" outdate="100"/>
    </server>
    <server name="call">
      <rule position="7" file_path="records" file_mask="*.meta" outdate="90"/>
      <rule position="8" file_path="records" file_mask="*.pcm" outdate="90"/>
    </server>
  </request>
</in>

```

Ответ:

```

status: 201
content:

```

Нс_restfs_domain_autocleaner_change_rule

Команда предназначена для изменения определенных правил автоматической подчистки устаревших файлов на выбранном домене.

Аналог команды в CoCon:

domain/<DOMAIN-NAME>/restfs/autocleaner/change-rule

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_domain_autocleaner_change_rule

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="server" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="rule" minOccurs="1" maxOccurs="unbound
eInteger" use="required"/>
                    <xs:attribute name="position" type="xs:nonNegativ
use="optional"/>
                    <xs:attribute name="file_path" type="xs:string"
use="optional"/>
                    <xs:attribute name="file_mask" type="xs:string"
Integer" use="optional"/>
                    <xs:attribute name="outdate" type="xs:nonNegative
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="name" type="serverType" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="domain" type="xs:string" use="required"/>
    </xs:complexType>
  </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:simpleType name="serverType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="fax"/>
    <xs:enumeration value="call"/>
    <xs:enumeration value="sounds"/>
    <xs:enumeration value="voicemail"/>
  </xs:restriction>
</xs:simpleType>

```



```

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/restfs_domain_autocleaner_change_rule

```

<in>
  <request domain="biysk.local">
    <server name="fax">
      <rule position="1" file_path="fax" file_mask="*.png" outdate="100"/>
    </server>
  </request>
</in>

```

Ответ:

```

status: 201
content:

```

[Hc_restfs_domain_autocleaner_remove_rule](#)

Команда предназначена для удаления определенных правил автоматической подчистки устаревших файлов на выбранном домене.

Аналог команды в CoCon:

domain/<DOMAIN-NAME>/restfs/autocleaner/remove-rule

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_domain_autocleaner_remove_rule

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="server" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="rule" minOccurs="1" maxOccurs="unbound
d">
                      <xs:complexType>
                        <xs:attribute name="position" type="xs:nonNegativ
eInteger" use="required"/>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
                <xs:attribute name="name" type="serverType" use="required"/>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
          <xs:attribute name="domain" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

  <xs:simpleType name="serverType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="fax"/>
      <xs:enumeration value="call"/>
      <xs:enumeration value="sounds"/>
      <xs:enumeration value="voicemail"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="server" minOccurs="0" maxOccurs="unbounded">
          <xs:complexType>

```

```

<xs:sequence>
  <xs:element name="start" minOccurs="1" maxOccurs="1">
    <xs:complexType>
      <xs:attribute name="at" type="xs:string" use="required"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="rule" minOccurs="0" maxOccurs="unbounded">
    <xs:complexType>
      <xs:attribute name="position" type="xs:integer" use="required"
"/>
      <xs:attribute name="file_path" type="xs:string" use="required"
"/>
      <xs:attribute name="file_mask" type="xs:string" use="required"
"/>
      <xs:attribute name="outdate" type="xs:string" use="required"/
>
    </xs:complexType>
  </xs:element>
</xs:sequence>
<xs:attribute name="name" type="xs:string" use="required"/>
<xs:attribute name="peer" type="xs:string" use="required"/>
</xs:complexType>
</xs:element>
<xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/restfs_domain_autocleaner_remove_rule

```

<in>
  <request domain="biysk.local">
    <server name="fax">
      <rule position="6"/>
    </server/>
    <server name="call">
      <rule position="6"/>
      <rule position="22"/>
    </server>
  </request>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_restfs_
domain_autocleaner_remove_rule.xsd">
  <error cmd="hc_restfs_domain_autocleaner_remove_rule" reason="high id: 22" entity="call"/>
</out>
```

hc_restfs_domain_autocleaner_info

Команда предназначена для просмотра списка правил автоподчистки на определенном домене.

Аналог команды в CoCon:

domain/<DOMAIN-NAME>/restfs/autocleaner/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_domain_autocleaner_info

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="server" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="name" type="serverType" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:simpleType name="serverType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="fax"/>
      <xs:enumeration value="call"/>
      <xs:enumeration value="sounds"/>
      <xs:enumeration value="voicemail"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="server" minOccurs="0" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="rule" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="position" type="xs:integer" use="required"
"/>
                  <xs:attribute name="file_path" type="xs:string" use="required"
"/>
                  <xs:attribute name="file_mask" type="xs:string" use="required"
"/>

```

```

        <xs:attribute name="outdate" type="xs:string" use="required"/
    >
        </xs:complexType>
    </xs:element>
</xs:sequence>
<xs:attribute name="name" type="xs:string" use="required"/>
<xs:attribute name="peer" type="xs:string" use="required"/>
<xs:attribute name="url" type="xs:string" use="required"/>
</xs:complexType>
</xs:element>
<xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/restfs_domain_autocleaner_info

```

<in>
  <request domain="biysk.local">
    <server name="call"/>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_restfs_
domain_autocleaner_info.xsd">
  <server name="call" peer="default" url="http://system.restfs.ecss:9990">
    <rule position="1" file_path="fax" file_mask="*.pdf" outdate="180"/>
    <rule position="2" file_path="records" file_mask="*.mp3" outdate="180"/>
    <rule position="3" file_path="records" file_mask="*.meta" outdate="180"/>
    <rule position="4" file_path="records" file_mask="*.pcm" outdate="180"/>
    <rule position="5" file_path="records" file_mask="*.wav" outdate="180"/>
  </server>
</out>

```

hc_restfs_domain_autocleaner_set

Команда предназначена для изменения настроек автоподчистки на определенном домене.

Аналог команды в CoCon:

domain/<DOMAIN-NAME>/restfs/autocleaner/change-rule

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_domain_autocleaner_set

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="server" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="serverType" use="required"/>
          <xs:attribute name="hh" type="xs:nonNegativeInteger" use="required"/>
          <xs:attribute name="mm" type="xs:nonNegativeInteger" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:simpleType name="serverType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="fax"/>
      <xs:enumeration value="call"/>
      <xs:enumeration value="sounds"/>
      <xs:enumeration value="voicemail"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```


Hc_restfs_domain_list - Просмотр списка файлов по указанному пути на определенном домене

[Hc_restfs_domain_list](#)

Команда предназначена для просмотра списка файлов по указанному пути на определенном домене.

Аналог команды в CoCon:

`domain/<DOMAIN-NAME>/restfs/List`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_domain_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="server" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="name" type="serverType" use="required"/>
                  <xs:attribute name="path" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:simpleType name="serverType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="fax"/>
      <xs:enumeration value="call"/>
      <xs:enumeration value="sounds"/>
      <xs:enumeration value="voicemail"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="clusterType">
    <xs:sequence>
      <xs:element name="file" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="type" type="xs:string" use="required"/>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="size" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- out -->

```

```

<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="server" type="clusterType" minOccurs="0" maxOccurs="unbounded"/
    >
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/restfs_domain_list

```

<in>
  <request domain="DOMAIN">
    <server name="cluster2" path=""/>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_restfs_
domain_list.xsd">
  <server name="call">
    <file type="directory" name="records" size="0"/>
    <file type="directory" name="subscribers" size="0"/>
    <file type="regular" name="cleaner.json" size="337"/>
  </server>
</out>

```

Нс_restfs_list_clusters - Просмотр списка файлов на кластере Restfs**Нс_restfs_list_clusters**

Команда предназначена для получения списка файлов, хранящихся на файловой системе RestFS.

Аналог команды в CoCon:

restfs/<RESTFS-CLUSTERNAME>/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_list_clusters

Код HTTP-ответа:

- 200 – в случае успеха;
- 434 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="cluster" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="name" type="xs:string" use="required"/>
                  <xs:attribute name="path" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:complexType name="clusterType">
    <xs:sequence>
      <xs:element name="file" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="type" type="xs:string" use="required"/>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="size" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="cluster" type="clusterType" minOccurs="0" maxOccurs="unbounded"
/>
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```
</xs:schema>
```

Пример:

http://host:port/commands/restfs_list_clusters

```
<in>  
  <request>  
    <cluster name="default" path="/domain"/>  
    <cluster name="default" path="domain"/>  
    <cluster name="default" path="system/sounds"/>  
  </request>  
</in>
```

Ответ:

```
status: 200
content:
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_restfs_
list_clusters.xsd">
  <cluster name="default">
    <file type="directory" name="curency" size="0"/>
    <file type="directory" name="dow" size="0"/>
    <file type="directory" name="enumeration" size="0"/>
    <file type="directory" name="et" size="0"/>
    <file type="directory" name="numbers" size="0"/>
    <file type="directory" name="prompts" size="0"/>
    <file type="directory" name="push_numbers" size="0"/>
    <file type="directory" name="queue" size="0"/>
    <file type="directory" name="variable" size="0"/>
    <file type="directory" name="voicemail" size="0"/>
    <file type="regular" name="activated.wav" size="9562"/>
    <file type="regular" name="ai_activated.wav" size="28752"/>
    <file type="regular" name="ai_added.wav" size="19270"/>
    <file type="regular" name="ai_agent_already_on.wav" size="214020"/>
    <file type="regular" name="ai_agent_auxwork.wav" size="282310"/>
    <file type="regular" name="ai_agent_busy.wav" size="260292"/>
    <file type="regular" name="ai_agent_logged_out.wav" size="38078"/>
    <file type="regular" name="ai_agent_login_ok.wav" size="31808"/>
    <file type="regular" name="ai_agent_number_occupied.wav" size="264004"/>
    <file type="regular" name="ai_alarm_multi_digits.wav" size="32526"/>
    <file type="regular" name="ai_alarm_off.wav" size="32720"/>
    <file type="regular" name="ai_alarm_off_today.wav" size="40746"/>
    <file type="regular" name="ai_alarm_on.wav" size="31048"/>
    <file type="regular" name="ai_alarm_one_digit.wav" size="24584"/>
    <file type="regular" name="ai_alerting.wav" size="80078"/>
    <file type="regular" name="ai_alerting_nd.wav" size="64298"/>
    <file type="regular" name="ai_already_activated.wav" size="195296"/>
    <file type="regular" name="ai_auth_incorrect.wav" size="28712"/>
    <file type="regular" name="ai_badconnection.wav" size="34096"/>
    <file type="regular" name="ai_badroute.wav" size="43034"/>
    <file type="regular" name="ai_busy.wav" size="11278"/>
    <file type="regular" name="ai_busy_long.wav" size="58638"/>
    <file type="regular" name="ai_busy_overload.wav" size="5678"/>
    <file type="regular" name="ai_call_forwarding.wav" size="31466"/>
    <file type="regular" name="ai_call_recording_announcement.wav" size="184844"/>
    <file type="regular" name="ai_call_recording_notification.wav" size="240926"/>
    <file type="regular" name="ai_changed.wav" size="38948"/>
    <file type="regular" name="ai_check_balance.wav" size="154782"/>
    <file type="regular" name="ai_conference_can_not_be_extended.wav" size="46034"/>
    <file type="regular" name="ai_conference_destroyed.wav" size="42342"/>
    <file type="regular" name="ai_conference_full.wav" size="46034"/>
    <file type="regular" name="ai_conference_has_been_extended.wav" size="35818"/>
```

```
<file type="regular" name="ai_conference_prompt.wav" size="38326"/>
<file type="regular" name="ai_connection_not_available_now.wav" size="31480"/>
<file type="regular" name="ai_deactivated.wav" size="33978"/>
<file type="regular" name="ai_deactivated_all.wav" size="48978"/>
<file type="regular" name="ai_debt.wav" size="43662"/>
<file type="regular" name="ai_empty.wav" size="57352"/>
<file type="regular" name="ai_enter_number_of_conference.wav" size="17270"/>
<file type="regular" name="ai_enter_number_sharp.wav" size="62990"/>
<file type="regular" name="ai_enter_password.wav" size="12036"/>
<file type="regular" name="ai_enter_pin.wav" size="23882"/>
<file type="regular" name="ai_enter_pin_number_sharp.wav" size="91088"/>
<file type="regular" name="ai_enter_subscriber_prompt_number.wav" size="43184"/>
<file type="regular" name="ai_error.wav" size="27572"/>
<file type="regular" name="ai_every1.wav" size="18928"/>
<file type="regular" name="ai_every2.wav" size="20600"/>
<file type="regular" name="ai_every3.wav" size="16628"/>
<file type="regular" name="ai_every4.wav" size="17238"/>
<file type="regular" name="ai_exact_time.wav" size="10106"/>
<file type="regular" name="ai_exit.wav" size="46568"/>
<file type="regular" name="ai_for_help_at_any_time.wav" size="38772"/>
<file type="regular" name="ai_from.wav" size="13912"/>
<file type="regular" name="ai_incomplete.wav" size="27036"/>
<file type="regular" name="ai_intercom_announce.wav" size="318306"/>
<file type="regular" name="ai_intercom_end.wav" size="380868"/>
<file type="regular" name="ai_invalid_number_of_conference.wav" size="24376"/>
<file type="regular" name="ai_invalidnumber.wav" size="27036"/>
<file type="regular" name="ai_last_incoming_number.wav" size="51486"/>
<file type="regular" name="ai_local_hold.wav" size="80078"/>
<file type="regular" name="ai_mailbox_empty.wav" size="52510"/>
<file type="regular" name="ai_my_number.wav" size="20976"/>
<file type="regular" name="ai_nearest1.wav" size="20182"/>
<file type="regular" name="ai_nearest2.wav" size="20182"/>
<file type="regular" name="ai_nearest3.wav" size="22270"/>
<file type="regular" name="ai_nearest4.wav" size="23942"/>
<file type="regular" name="ai_no_answer.wav" size="36858"/>
<file type="regular" name="ai_no_money_to_call.wav" size="210878"/>
<file type="regular" name="ai_not_reachable.wav" size="37254"/>
<file type="regular" name="ai_notaccess.wav" size="65234"/>
<file type="regular" name="ai_notconnected.wav" size="50088"/>
<file type="regular" name="ai_notification.wav" size="83276"/>
<file type="regular" name="ai_notpay.wav" size="38808"/>
<file type="regular" name="ai_number_absent.wav" size="55342"/>
<file type="regular" name="ai_number_incorrect.wav" size="20824"/>
<file type="regular" name="ai_number_is_hidden.wav" size="18734"/>
<file type="regular" name="ai_number_not_detected.wav" size="18734"/>
<file type="regular" name="ai_number_not_found.wav" size="66114"/>
<file type="regular" name="ai_number_not_ready.wav" size="52028"/>
<file type="regular" name="ai_one_minute_cost.wav" size="195404"/>
<file type="regular" name="ai_out_of_order.wav" size="38112"/>
<file type="regular" name="ai_overload.wav" size="39366"/>
```



```

<file type="regular" name="ai_park_background_music.wav" size="3892940"/>
<file type="regular" name="ai_park_slot.wav" size="15180"/>
<file type="regular" name="ai_park_slot_busy.wav" size="12254"/>
<file type="regular" name="ai_park_slot_empty.wav" size="41456"/>
<file type="regular" name="ai_park_slot_number.wav" size="24794"/>
<file type="regular" name="ai_participant_of_conference.wav" size="28154"/>
<file type="regular" name="ai_pin_accepted.wav" size="25572"/>
<file type="regular" name="ai_pin_incorrect.wav" size="33932"/>
<file type="regular" name="ai_please_enter_number.wav" size="30172"/>
<file type="regular" name="ai_prompt_recorder_intro.wav" size="116320"/>
<file type="regular" name="ai_prompt_recorder_intro2.wav" size="164906"/>
<file type="regular" name="ai_remote_hold.wav" size="3892940"/>
<file type="regular" name="ai_ringback_second.wav" size="546266"/>
<file type="regular" name="ai_rubles.wav" size="92684"/>
<file type="regular" name="ai_silent.wav" size="12302"/>
<file type="regular" name="ai_stay_on_line.wav" size="389572"/>
<file type="regular" name="ai_subscriber_busy.wav" size="24318"/>
<file type="regular" name="ai_supervising_mode_conference.wav" size="40446"/>
<file type="regular" name="ai_supervising_mode_consult.wav" size="45230"/>
<file type="regular" name="ai_supervising_mode_observing.wav" size="39964"/>
<file type="regular" name="ai_supervising_mode_selected.wav" size="43332"/>
<file type="regular" name="ai_system_problem.wav" size="39366"/>
<file type="regular" name="ai_tempdenied.wav" size="37254"/>
<file type="regular" name="ai_timeout.wav" size="35122"/>
<file type="regular" name="ai_to_go_back_at_any_time.wav" size="58968"/>
<file type="regular" name="ai_to_make_alarm_intro.wav" size="49962"/>
<file type="regular" name="ai_to_make_subscriber_intro.wav" size="21868"/>
<file type="regular" name="ai_to_make_voicemail_intro.wav" size="58270"/>
<file type="regular" name="ai_to_turn_off.wav" size="28540"/>
<file type="regular" name="ai_to_turn_on.wav" size="23106"/>
<file type="regular" name="ai_unavailable.wav" size="28916"/>
<file type="regular" name="ai_until.wav" size="14330"/>
<file type="regular" name="ai_wait_connection.wav" size="64024"/>
<file type="regular" name="ai_wait_time.wav" size="25342"/>
<file type="regular" name="ai_wrong_number.wav" size="27036"/>
<file type="regular" name="ai_you.wav" size="11670"/>
<file type="regular" name="ai_you_are_first_participant.wav" size="50806"/>
<file type="regular" name="ai_your_balance_is.wav" size="53804"/>
<file type="regular" name="conference_already_exists.wav" size="246100"/>
<file type="regular" name="conference_not_exists.wav" size="204704"/>
<file type="regular" name="for_call_to_number.wav" size="151784"/>
<file type="regular" name="reached_conference_restriction.wav" size="258330"/>
<file type="regular" name="to_number.wav" size="107332"/>
</cluster>
<cluster name="default">
  <file type="directory" name="biysk.local" size="0"/>
</cluster>
<cluster name="default">
  <file type="directory" name="biysk.local" size="0"/>
</cluster>

```

```
</out>
```

Hc_restfs_list - Просмотр списка файлов на RestFS для системы или виртуальной АТС

- [Hc_restfs_list](#)
- [Hc_restfs_url](#)

[Hc_restfs_list](#)

Команда предназначена для получения списка файлов, хранящихся на файловой системе RestFS на уровне системы или виртуальной АТС. За одну операцию можно получить список файлов для нескольких папок.

Аналог команды в CoCon:

`domain/<DOMAIN>/restfs/list`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/restfs_list

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- systemRequestType -->
  <xs:complexType name="systemRequestType">
    <xs:attribute name="source" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- domainRequestType -->
  <xs:complexType name="domainRequestType">
    <xs:attribute name="source" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="cause" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- fileType -->
  <xs:complexType name="fileType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="size" type="xs:integer" use="required"/>
  </xs:complexType>
  <!-- groupType -->
  <xs:complexType name="groupType">
    <xs:sequence>
      <xs:element name="file" type="fileType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- systemResultType -->
  <xs:complexType name="systemResultType">
    <xs:sequence>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="1"/>
      <xs:element name="group" type="groupType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="file" type="fileType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="source" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- domainResultType -->
  <xs:complexType name="domainResultType">
    <xs:sequence>
      <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="1"/>
      <xs:element name="group" type="groupType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="file" type="fileType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="source" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>

```

```

        <xs:element name="system" type="systemRequestType" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="domain" type="domainRequestType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
</xs:complexType>
<!-- resultType -->
<xs:complexType name="resultType">
    <xs:sequence>
        <xs:element name="system" type="systemResultType" minOccurs="0" maxOccurs="unbounded"
/>
        <xs:element name="domain" type="domainResultType" minOccurs="0" maxOccurs="unbounded"
/>
    </xs:sequence>
</xs:complexType>
<!-- in -->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request" type="requestType"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="result" type="resultType"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/restfs_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="restfs_list.xsd">
  <request domain="biysk.local">
    <system source="sounds"/>
    <domain source="sounds"/>
  </request>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="restfs_list.xsd">
  <result>
    <system source="sounds">
      <group name="currency">
        <file name="cent.wav" size="95336"/>
        <file name="cent_a.wav" size="96512"/>
        <file name="cent_ov.wav" size="105450"/>
        <file name="euro_cent.wav" size="123324"/>
        <file name="euro_cent_a.wav" size="118150"/>
        <file name="euro_cent_ov.wav" size="136496"/>
        <file name="kopecks.wav" size="9274"/>
        <file name="ruble.wav" size="5972"/>
        <file name="ruble_a.wav" size="6762"/>
        <file name="ruble_ov.wav" size="8458"/>
      </group>
      <group name="dow">
        <file name="ai_friday2.wav" size="7866"/>
        <file name="ai_saturday2.wav" size="8494"/>
        <file name="ai_wednesday2.wav" size="9538"/>
        <file name="friday.wav" size="8076"/>
        <file name="monday.wav" size="9120"/>
        <file name="saturday.wav" size="8702"/>
        <file name="sunday.wav" size="10584"/>
        <file name="thursday.wav" size="9120"/>
        <file name="tuesday.wav" size="9120"/>
        <file name="wednesday.wav" size="8494"/>
      </group>
      <group name="enumeration">
        <file name="h-1.wav" size="8404"/>
        <file name="h-10.wav" size="13418"/>
        <file name="h-11.wav" size="15926"/>
        <file name="h-12.wav" size="16762"/>
        <file name="h-13.wav" size="15926"/>
        <file name="h-14.wav" size="18016"/>
        <file name="h-15.wav" size="15926"/>
        <file name="h-16.wav" size="18016"/>
        <file name="h-2.wav" size="10076"/>
        <file name="h-3.wav" size="10910"/>
        <file name="h-4.wav" size="14672"/>
        <file name="h-5.wav" size="9240"/>
        <file name="h-6.wav" size="12164"/>
        <file name="h-7.wav" size="13000"/>
        <file name="h-8.wav" size="13000"/>
        <file name="h-9.wav" size="14254"/>
      </group>
    </system>
  </group>
</out>
```

```
<file name="1.wav" size="12282"/>
<file name="10.wav" size="12002"/>
<file name="11.wav" size="13532"/>
<file name="12.wav" size="13864"/>
<file name="13.wav" size="13326"/>
<file name="14.wav" size="16392"/>
<file name="15.wav" size="13186"/>
<file name="16.wav" size="14254"/>
<file name="17.wav" size="14998"/>
<file name="18.wav" size="17654"/>
<file name="19.wav" size="16972"/>
<file name="1h.wav" size="10186"/>
<file name="2.wav" size="11312"/>
<file name="20.wav" size="14812"/>
<file name="2h.wav" size="11354"/>
<file name="3.wav" size="8590"/>
<file name="30.wav" size="11338"/>
<file name="4.wav" size="13814"/>
<file name="40.wav" size="13372"/>
<file name="5.wav" size="9798"/>
<file name="50.wav" size="13140"/>
<file name="6.wav" size="12118"/>
<file name="7.wav" size="11608"/>
<file name="8.wav" size="12954"/>
<file name="9.wav" size="12584"/>
<file name="exact_time.wav" size="19618"/>
<file name="hours1.wav" size="11296"/>
<file name="hours2.wav" size="14224"/>
<file name="hours3.wav" size="14304"/>
<file name="minute1.wav" size="10656"/>
<file name="minute2.wav" size="13350"/>
<file name="minute3.wav" size="14568"/>
<file name="minute4.wav" size="13012"/>
<file name="second1.wav" size="12672"/>
<file name="second2.wav" size="11000"/>
<file name="second3.wav" size="9956"/>
</group>
<group name="numbers">
  <file name="#.wav" size="13696"/>
  <file name="#1.wav" size="13754"/>
  <file name="*.wav" size="15882"/>
  <file name="*1.wav" size="80438"/>
  <file name="0.wav" size="11654"/>
  <file name="1.wav" size="9728"/>
  <file name="1n.wav" size="6376"/>
  <file name="2.wav" size="11354"/>
  <file name="3.wav" size="8590"/>
  <file name="4.wav" size="13814"/>
  <file name="5.wav" size="9798"/>
  <file name="6.wav" size="12118"/>
```

```
<file name="7.wav" size="11608"/>
<file name="8.wav" size="12954"/>
<file name="9.wav" size="12584"/>
<file name="A.wav" size="69582"/>
<file name="B.wav" size="68002"/>
<file name="C.wav" size="75932"/>
<file name="D.wav" size="75580"/>
<file name="E.wav" size="67586"/>
<file name="F.wav" size="84514"/>
</group>
<group name="prompts">
  <file name="voicemail_intro.wav" size="626066"/>
</group>
<group name="push_numbers">
  <file name="ai_push_#.wav" size="12056"/>
  <file name="ai_push_*.wav" size="13044"/>
  <file name="ai_push_0.wav" size="15180"/>
  <file name="ai_push_1.wav" size="14442"/>
  <file name="ai_push_2.wav" size="15180"/>
  <file name="ai_push_3.wav" size="15998"/>
  <file name="ai_push_4.wav" size="15980"/>
  <file name="ai_push_5.wav" size="16060"/>
  <file name="ai_push_6.wav" size="17454"/>
  <file name="ai_push_7.wav" size="16156"/>
  <file name="ai_push_8.wav" size="17986"/>
  <file name="ai_push_9.wav" size="16064"/>
</group>
<group name="queue">
  <file name="ai_do_you_want_decline_callback.wav" size="48688"/>
  <file name="ai_do_you_want_us_to_call_you_back.wav" size="43254"/>
  <file name="ai_limit_reached.wav" size="59972"/>
  <file name="ai_notify_callback.wav" size="43254"/>
  <file name="ai_now_you_can_hang_up.wav" size="64152"/>
  <file name="ai_please_rate_us.wav" size="283026"/>
  <file name="ai_queue_enqueued.wav" size="30144"/>
  <file name="ai_queue_full.wav" size="31480"/>
  <file name="ai_queue_greeting.wav" size="52594"/>
  <file name="ai_queue_music.wav" size="3892940"/>
  <file name="ai_queue_position.wav" size="16226"/>
  <file name="ai_queue_predicted_time.wav" size="22912"/>
  <file name="ai_rate_from_0_to_9.wav" size="393100"/>
  <file name="ai_rate_from_1_to_5.wav" size="383220"/>
  <file name="ai_rate_us_from_0_to_1.wav" size="480124"/>
  <file name="ai_thanks_you_for_your_opinion.wav" size="203998"/>
  <file name="ai_you_can_hang_up.wav" size="58718"/>
  <file name="ai_you_can_still_wait.wav" size="105112"/>
  <file name="ai_your_opinion_is_very_important.wav" size="262564"/>
</group>
<group name="variable"/>
<group name="voicemail">
```

```
<file name="ai_call_to_sender.wav" size="39826"/>
<file name="ai_change_service_settings.wav" size="44840"/>
<file name="ai_confirm_remove_messages.wav" size="40746"/>
<file name="ai_decline_remove_messages.wav" size="26118"/>
<file name="ai_enter_current_password.wav" size="41498"/>
<file name="ai_enter_new_password.wav" size="38572"/>
<file name="ai_enter_number_sharp.wav" size="59688"/>
<file name="ai_exit.wav" size="57440"/>
<file name="ai_for_exit_from_service.wav" size="25420"/>
<file name="ai_for_listen_next_message.wav" size="23122"/>
<file name="ai_for_record_voice_message_push.wav" size="29600"/>
<file name="ai_leave_message_after_beep.wav" size="218156"/>
<file name="ai_listen_current_file.wav" size="33404"/>
<file name="ai_listen_file.wav" size="45756"/>
<file name="ai_listen_files.wav" size="22076"/>
<file name="ai_listen_messages.wav" size="31618"/>
<file name="ai_listen_new_messages.wav" size="19360"/>
<file name="ai_listen_old_messages.wav" size="21868"/>
<file name="ai_listen_prompt.wav" size="21032"/>
<file name="ai_listen_save_messages.wav" size="47348"/>
<file name="ai_message_from_hidden_number.wav" size="55840"/>
<file name="ai_message_from_number.wav" size="21240"/>
<file name="ai_message_saved.wav" size="16852"/>
<file name="ai_message_successfully_removed.wav" size="17480"/>
<file name="ai_messages_successfully_removed.wav" size="17898"/>
<file name="ai_new_message.wav" size="14554"/>
<file name="ai_new_messages_1.wav" size="13298"/>
<file name="ai_new_messages_2.wav" size="14136"/>
<file name="ai_no_more_messages.wav" size="15180"/>
<file name="ai_old_message.wav" size="15808"/>
<file name="ai_old_messages_1.wav" size="13358"/>
<file name="ai_old_messages_2.wav" size="16852"/>
<file name="ai_participant_can_not_answer.wav" size="366414"/>
<file name="ai_password_successfully_changed.wav" size="28958"/>
<file name="ai_passwords_different.wav" size="36900"/>
<file name="ai_pause_resume_message.wav" size="69500"/>
<file name="ai_play_back_message.wav" size="23540"/>
<file name="ai_play_forward_message.wav" size="23540"/>
<file name="ai_play_message_details.wav" size="44422"/>
<file name="ai_reenter_new_password.wav" size="37736"/>
<file name="ai_remove_current_message.wav" size="22076"/>
<file name="ai_remove_new_messages.wav" size="21450"/>
<file name="ai_remove_old_messages.wav" size="25002"/>
<file name="ai_remove_save_messages.wav" size="52364"/>
<file name="ai_replay_message.wav" size="25420"/>
<file name="ai_save_current_message.wav" size="39482"/>
<file name="ai_save_file.wav" size="48630"/>
<file name="ai_saved_message.wav" size="34810"/>
<file name="ai_saved_messages_1.wav" size="34810"/>
<file name="ai_saved_messages_2.wav" size="34810"/>
```



```
<file name="ai_say_message_after_beep.wav" size="63276"/>
<file name="ai_skip_message.wav" size="50274"/>
<file name="ai_ss_unavailable.wav" size="14554"/>
<file name="ai_to_remove_message.wav" size="190606"/>
<file name="ai_to_remove_subscriber_intro.wav" size="45676"/>
<file name="ai_to_remove_voicemail_intro.wav" size="63604"/>
<file name="ai_try_once_more.wav" size="16524"/>
<file name="ai_voicemail_change_common_settings.wav" size="51110"/>
<file name="ai_voicemail_change_password.wav" size="32302"/>
<file name="ai_voicemail_change_play_details.wav" size="74098"/>
<file name="ai_voicemail_change_send_by_email.wav" size="78696"/>
<file name="ai_voicemail_change_write_by_busy.wav" size="82874"/>
<file name="ai_voicemail_change_write_by_no_reply.wav" size="74516"/>
<file name="ai_voicemail_change_write_by_out_of_service.wav" size="85800"/>
<file name="ai_voicemail_change_write_by_unconditional.wav" size="71172"/>
<file name="ai_voicemail_change_write_conditions.wav" size="64066"/>
<file name="ai_voicemail_mailbox_empty.wav" size="31898"/>
<file name="ai_voicemail_mgm_intro.wav" size="49856"/>
<file name="ai_voicemail_password_incorrect.wav" size="33556"/>
<file name="ai_voicemail_remove_user_intro.wav" size="42750"/>
<file name="ai_voicemail_write_user_intro.wav" size="18734"/>
<file name="ai_you_have.wav" size="8494"/>
</group>
<file name="activated.wav" size="9562"/>
<file name="ai_activated.wav" size="28752"/>
<file name="ai_added.wav" size="19270"/>
<file name="ai_agent_already_on.wav" size="214020"/>
<file name="ai_agent_auxwork.wav" size="282310"/>
<file name="ai_agent_busy.wav" size="260292"/>
<file name="ai_agent_logged_out.wav" size="38078"/>
<file name="ai_agent_login_ok.wav" size="31808"/>
<file name="ai_agent_number_occupied.wav" size="264004"/>
<file name="ai_alarm_multi_digits.wav" size="32526"/>
<file name="ai_alarm_off.wav" size="32720"/>
<file name="ai_alarm_off_today.wav" size="40746"/>
<file name="ai_alarm_on.wav" size="31048"/>
<file name="ai_alarm_one_digit.wav" size="24584"/>
<file name="ai_alerting.wav" size="80078"/>
<file name="ai_alerting_nd.wav" size="64298"/>
<file name="ai_already_activated.wav" size="195296"/>
<file name="ai_auth_incorrect.wav" size="28712"/>
<file name="ai_badconnection.wav" size="34096"/>
<file name="ai_badroute.wav" size="43034"/>
<file name="ai_busy.wav" size="11278"/>
<file name="ai_busy_long.wav" size="58638"/>
<file name="ai_busy_overload.wav" size="5678"/>
<file name="ai_call_forwarding.wav" size="31466"/>
<file name="ai_call_recording_announcement.wav" size="184844"/>
<file name="ai_call_recording_notification.wav" size="240926"/>
<file name="ai_changed.wav" size="38948"/>
```

```
<file name="ai_check_balance.wav" size="154782"/>
<file name="ai_conference_can_not_be_extended.wav" size="46034"/>
<file name="ai_conference_destroyed.wav" size="42342"/>
<file name="ai_conference_full.wav" size="46034"/>
<file name="ai_conference_has_been_extended.wav" size="35818"/>
<file name="ai_conference_prompt.wav" size="38326"/>
<file name="ai_connection_not_available_now.wav" size="31480"/>
<file name="ai_deactivated.wav" size="33978"/>
<file name="ai_deactivated_all.wav" size="48978"/>
<file name="ai_debt.wav" size="43662"/>
<file name="ai_empty.wav" size="57352"/>
<file name="ai_enter_number_of_conference.wav" size="17270"/>
<file name="ai_enter_number_sharp.wav" size="62990"/>
<file name="ai_enter_password.wav" size="12036"/>
<file name="ai_enter_pin.wav" size="23882"/>
<file name="ai_enter_pin_number_sharp.wav" size="91088"/>
<file name="ai_enter_subscriber_prompt_number.wav" size="43184"/>
<file name="ai_error.wav" size="27572"/>
<file name="ai_every1.wav" size="18928"/>
<file name="ai_every2.wav" size="20600"/>
<file name="ai_every3.wav" size="16628"/>
<file name="ai_every4.wav" size="17238"/>
<file name="ai_exact_time.wav" size="10106"/>
<file name="ai_exit.wav" size="46568"/>
<file name="ai_for_help_at_any_time.wav" size="38772"/>
<file name="ai_from.wav" size="13912"/>
<file name="ai_incomplete.wav" size="27036"/>
<file name="ai_intercom_announce.wav" size="318306"/>
<file name="ai_intercom_end.wav" size="380868"/>
<file name="ai_invalid_number_of_conference.wav" size="24376"/>
<file name="ai_invalidnumber.wav" size="27036"/>
<file name="ai_last_incoming_number.wav" size="51486"/>
<file name="ai_local_hold.wav" size="80078"/>
<file name="ai_mailbox_empty.wav" size="52510"/>
<file name="ai_my_number.wav" size="20976"/>
<file name="ai_nearest1.wav" size="20182"/>
<file name="ai_nearest2.wav" size="20182"/>
<file name="ai_nearest3.wav" size="22270"/>
<file name="ai_nearest4.wav" size="23942"/>
<file name="ai_no_answer.wav" size="36858"/>
<file name="ai_no_money_to_call.wav" size="210878"/>
<file name="ai_not_reachable.wav" size="37254"/>
<file name="ai_notaccess.wav" size="65234"/>
<file name="ai_notconnected.wav" size="50088"/>
<file name="ai_notification.wav" size="83276"/>
<file name="ai_notpay.wav" size="38808"/>
<file name="ai_number_absent.wav" size="55342"/>
<file name="ai_number_incorrect.wav" size="20824"/>
<file name="ai_number_is_hidden.wav" size="18734"/>
<file name="ai_number_not_detected.wav" size="18734"/>
```

```
<file name="ai_number_not_found.wav" size="66114"/>
<file name="ai_number_not_ready.wav" size="52028"/>
<file name="ai_one_minute_cost.wav" size="195404"/>
<file name="ai_out_of_order.wav" size="38112"/>
<file name="ai_overload.wav" size="39366"/>
<file name="ai_park_background_music.wav" size="3892940"/>
<file name="ai_park_slot.wav" size="15180"/>
<file name="ai_park_slot_busy.wav" size="12254"/>
<file name="ai_park_slot_empty.wav" size="41456"/>
<file name="ai_park_slot_number.wav" size="24794"/>
<file name="ai_participant_of_conference.wav" size="28154"/>
<file name="ai_pin_accepted.wav" size="25572"/>
<file name="ai_pin_incorrect.wav" size="33932"/>
<file name="ai_please_enter_number.wav" size="30172"/>
<file name="ai_prompt_recorder_intro.wav" size="116320"/>
<file name="ai_prompt_recorder_intro2.wav" size="164906"/>
<file name="ai_remote_hold.wav" size="3892940"/>
<file name="ai_ringback_second.wav" size="546266"/>
<file name="ai_rubles.wav" size="92684"/>
<file name="ai_silent.wav" size="12302"/>
<file name="ai_stay_on_line.wav" size="389572"/>
<file name="ai_subscriber_busy.wav" size="24318"/>
<file name="ai_supervising_mode_conference.wav" size="40446"/>
<file name="ai_supervising_mode_consult.wav" size="45230"/>
<file name="ai_supervising_mode_observing.wav" size="39964"/>
<file name="ai_supervising_mode_selected.wav" size="43332"/>
<file name="ai_system_problem.wav" size="39366"/>
<file name="ai_tempdenied.wav" size="37254"/>
<file name="ai_timeout.wav" size="35122"/>
<file name="ai_to_go_back_at_any_time.wav" size="58968"/>
<file name="ai_to_make_alarm_intro.wav" size="49962"/>
<file name="ai_to_make_subscriber_intro.wav" size="21868"/>
<file name="ai_to_make_voicemail_intro.wav" size="58270"/>
<file name="ai_to_turn_off.wav" size="28540"/>
<file name="ai_to_turn_on.wav" size="23106"/>
<file name="ai_unavailable.wav" size="28916"/>
<file name="ai_until.wav" size="14330"/>
<file name="ai_wait_connection.wav" size="64024"/>
<file name="ai_wait_time.wav" size="25342"/>
<file name="ai_wrong_number.wav" size="27036"/>
<file name="ai_you.wav" size="11670"/>
<file name="ai_you_are_first_participant.wav" size="50806"/>
<file name="ai_your_balance_is.wav" size="53804"/>
<file name="conference_already_exists.wav" size="246100"/>
<file name="conference_not_exists.wav" size="204704"/>
<file name="for_call_to_number.wav" size="151784"/>
<file name="reached_conference_restriction.wav" size="258330"/>
<file name="to_number.wav" size="107332"/>
</system>
<domain source="sounds"/>
```

```
</result>  
</out>
```

hc_restfs_url

Команда предназначена для получения URL до файла, хранящегося на RestFS для доступа до него по http.

Аналог команды в CoCon:

/domain/<DOMAIN>/restfs/list

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/restfs_url

Код http ответа:

201 – в случае успеха;

404 – в случае ошибки.

Xsd схема xml файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- systemRequestType -->
  <xs:complexType name="systemRequestType">
    <xs:attribute name="id" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Request Id. In case of id present, it will be returned in the result.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="source" type="xs:string" use="required"/>
    <xs:attribute name="group" type="xs:string" use="optional"/>
    <xs:attribute name="file" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- domainRequestType -->
  <xs:complexType name="domainRequestType">
    <xs:attribute name="id" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Request Id. In case of id present, it will be returned in the result.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="source" type="xs:string" use="required"/>
    <xs:attribute name="group" type="xs:string" use="optional"/>
    <xs:attribute name="file" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- systemResultType -->
  <xs:complexType name="systemResultType">
    <xs:attribute name="id" type="xs:string" use="optional"/>
    <xs:attribute name="url" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- domainResultType -->
  <xs:complexType name="domainResultType">
    <xs:attribute name="id" type="xs:string" use="optional"/>
    <xs:attribute name="url" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="system" type="systemRequestType" minOccurs="0"
maxOccurs="unbounded"/>
      <xs:element name="domain" type="domainRequestType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

```

```

<!-- resultType -->
<xs:complexType name="resultType">
  <xs:sequence>
    <xs:element name="system" type="systemResultType" minOccurs="0" maxOccurs="unbounded"
  />
    <xs:element name="domain" type="domainResultType" minOccurs="0" maxOccurs="unbounded"
  />
  </xs:sequence>
</xs:complexType>
<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request" type="requestType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="result" type="resultType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/restfs_url

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="restfs_url.xsd">
  <request domain="biysk.local">
    <server name="call" path=""/>
    <server name="fax" path=""/>
  </request>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_restfs_domain_list.xsd">
  <server name="fax">
    <file type="directory" name="records" size="0"/>
    <file type="directory" name="subscribers" size="0"/>
    <file type="regular" name="cleaner.json" size="337"/>
  </server>
  <server name="call">
    <file type="directory" name="records" size="0"/>
    <file type="directory" name="subscribers" size="0"/>
    <file type="regular" name="cleaner.json" size="337"/>
  </server>
</out>
```

Нс_role - Команды управления ролями

- [Нс_role_add_group](#)
- [Нс_role_declare](#)
- [Нс_role_del_group](#)
- [Нс_role_properties_clean](#)
- [Нс_role_properties_info](#)
- [Нс_role_properties_set](#)
- [Нс_role_remove](#)
- [Нс_set_role_to_user](#)
- [Нс_del_role_from_user](#)

Нс_role_add_group

Команда добавляет выбранные группы к существующей CoCon роли через HTTP-терминал.

Аналог команды в CoCon:

cocon/role/add-group

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/role_add_group

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Add groups to exists role.
      Usage: http://server:9999/commands/role_add_group.
      Implementation: hc_role_add_group.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- groupType -->
  <xs:complexType name="groupType">
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- roleType -->
  <xs:complexType name="roleType">
    <xs:sequence>
      <xs:element name="group" type="groupType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="role" type="roleType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>

```



```

    <xs:choice>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/role_add_group

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_role_add_group.xsd">
  <request>
    <role name="role1">
      <group name="group1"/>
      <group name="group2"/>
    </role>
  </request>
</in>

```

Ответ:

```

status: 201
content:

```

[Hc_role_declare](#)

Команда предназначена для создания новой CoCon роли через HTTP-терминал.

Аналог команды в CoCon:

cocon/role/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/role_declare

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Declare role with specific name.
      Usage: http://server:9999/commands/role_declare.
      Implementation: hc_role_declare.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- propertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute type="xs:string" name="key" use="required"/>
    <xs:attribute type="xs:string" name="value" use="required"/>
  </xs:complexType>

  <!-- roleType -->
  <xs:complexType name="roleType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="role" type="roleType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>

```

```

        <xs:element name="error" type="errorType"/>
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/role_declare

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_role_declare.xsd">
  <request>
    <role name="admin1_role">
      <property key="key" value="value"/>
    </role>
    <role name="rolename3" />
  </request>
</in>

```

Ответ:

```

status: 201
content:

```

hc_role_del_group

Команда удаляет выбранные группы существующей CoCon роли через HTTP-терминал.

Аналог команды в CoCon:

cocon/role/del-group

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/role_del_group

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Delete groups from exists role.
      Usage: http://server:9999/commands/role_del_group.
      Implementation: hc_role_del_group.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- groupType -->
  <xs:complexType name="groupType">
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- roleType -->
  <xs:complexType name="roleType">
    <xs:sequence>
      <xs:element name="group" type="groupType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="role" type="roleType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>

```

```

    <xs:choice>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/role_del_group

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_role_del_group.xsd">
  <request>
    <role name="rolename3">
      <group name="group1"/>
      <group name="group2"/>
    </role>
  </request>
</in>

```

Ответ:

```

status: 201
content:

```

[Hc_role_properties_clean](#)

Команда удаляет выбранное свойство для существующей CoCon роли через HTTP-терминал.

Аналог команды в CoCon:

cocon/role/.properties/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/role_properties_clean

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Remove specified properties from exists role.
      Usage: http://server:9999/commands/role\_properties\_set.
      Implementation: hc_role_properties_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- keyType -->
  <xs:complexType name="keyTypeIn">
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- roleTypeIn -->
  <xs:complexType name="roleTypeIn">
    <xs:sequence>
      <xs:element name="key" type="keyTypeIn" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="role" type="roleTypeIn" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- propertyType -->
  <xs:complexType name="propertyTypeOut">
    <xs:attribute type="xs:string" name="key" use="optional"/>
    <xs:attribute type="xs:string" name="value" use="optional"/>
  </xs:complexType>

```

```

</xs:complexType>

<!-- roleTypeOut -->
<xs:complexType name="roleTypeOut">
  <xs:sequence>
    <xs:element name="property" type="propertyTypeOut" maxOccurs="unbounded" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>

<!-- resultType -->
<xs:complexType name="resultType">
  <xs:sequence>
    <xs:element name="role" type="roleTypeOut" maxOccurs="unbounded" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="result" type="resultType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/role_properties_clean

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_role_properties_clean.xsd">
  <request>
    <role name="rolename3">
      <key name="one" />
      <key name="two" />
      <key name="three" />
    </role>
  </request>
</in>

```

Ответ:

```

status: 201
content:

```

Нс_role_properties_info

Команда просмотра свойства CoCon роли через HTTP-терминал.

Аналог команды в CoCon:

cocon/role/.properties/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/role_properties_info

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Showing information about properties of entered role .
      Usage: http://server:9999/commands/role_properties_info.
      Implementation: hc_role_properties_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- keyType -->
  <xs:complexType name="keyType">
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- roleTypeIn -->
  <xs:complexType name="roleTypeIn">
    <xs:sequence>
      <xs:element name="key" type="keyType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="role" type="roleTypeIn" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- propertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute type="xs:string" name="key" use="optional"/>
    <xs:attribute type="xs:string" name="value" use="optional"/>
  </xs:complexType>

```

```

</xs:complexType>

<!-- roleTypeOut -->
<xs:complexType name="roleTypeOut">
  <xs:sequence>
    <xs:element name="property" type="propertyType" maxOccurs="unbounded" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>

<!-- resultType -->
<xs:complexType name="resultType">
  <xs:sequence>
    <xs:element name="role" type="roleTypeOut" maxOccurs="unbounded" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="result" type="resultType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/role_properties_info

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_role_properties_info.xsd">
  <request>
    <role name="cc-admin">
      <key name="access_tree" />
    </role>
    <role name="pbx-10X-admin"></role>
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_role_properties_info.xsd">
  <result>
    <role name="cc-admin">

```



```

uIjogInJvdXRpbmcubWfUyWdLciJ9LCB7ImLuZm8i0iB0cnVLLCAibGVhZiI6IHRydWUsICJuYW1lIjogImFjY2Vzc
19tYW5hZ2VyIiwgInRpdGxLIjogIkFjY2VzcyBtYW5hZ2VyIiwgImNyZWf0ZSI6IHRydWUsICJtb2RpZnki0iB0cnV
LLCAicmVtb3ZlIjogdHJlZSwgIl9pY29uIjogImFjY2Vzcy5tYW5hZ2VyIn0sIHsiaW5mbyI6IHRydWUsICJuYW1lI
jogInNzX2luc3RhbGwiLCAidG0bGU0i0iAiu1MgaW5zdGFsbCIsICJjcmVhdGU0i0iB0cnVLLCAibW9kaWZ5IjogdHJ
lZSwgInJlbW92ZSI6IHRydWUsICJjaGlsZHJlbiI6Ift7ImLuZm8i0iB0cnVLLCAibGVhZiI6IHRydWUsICJuYW1lI
jogIkZlYXRlcmVDb2RlcyIsICJ0aXRzZSI6ICJTuYBjb2RlcyIsICJjcmVhdGU0i0iB0cnVLLCAibW9kaWZ5IjogdHJ
lZSwgInJlbW92ZSI6IHRydWV9XSwgIl9pY29uIjogInNzLmLuc3RhbGwifSwgeyJpbmZvIjogdHJlZSwgImxlyWYi0
iB0cnVLLCAibmFtZSI6ICJhcHBfZG9jc19kb2NzIiwgInRpdGxLIjogIkRvY3VtZW50YXRpb24iLCAiY3JlYXRlIjo
gdHJlZSwgIm1vZGhmeSI6IHRydWUsICJyZW1vdmUi0iB0cnVLLCAiX2ljb24i0iAiYXBwX2RvY3MuZG9jcyJ9LCB7I
mluZm8i0iB0cnVLLCAibGVhZiI6IHRydWUsICJuYW1lIjogImdhdGV3YXlfbWfUyWdLciIsICJ0aXRzZSI6ICJHYXR
ld2F5IG1hbmFnZXIiLCAiY3JlYXRlIjogdHJlZSwgIm1vZGhmeSI6IHRydWUsICJyZW1vdmUi0iB0cnVLLCAiX2ljb
24i0iAiZ2F0ZXdheS5tYW5hZ2VyIn0sIHsiaW5mbyI6IHRydWUsICJsZWf0ZSI6IHRydWUsICJtb2RpZnki0iB0cnVLLCAic
mVtb3ZlIjogdHJlZSwgIl9pY29uIjogInRydW5rLnNob3cifSwgeyJpbmZvIjogdHJlZSwgIm5hbWUi0iAibWdtdF9
ob3N0X2luZm8iLCAidG0bGU0i0iAiu1NXIGhvc3RzIGluZm8iLCAiY3JlYXRlIjogdHJlZSwgIm1vZGhmeSI6IHRyd
WUsICJyZW1vdmUi0iB0cnVLLCAiY2hpbGRyZW4i0iBbeyJpbmZvIjogdHJlZSwgImxlyWYi0iB0cnVLLCAibmFtZSI6
ICJHcmFwaHMiLCAidG0bGU0i0iAiqWN0aXZlIGNhbGxzIiwgImNyZWf0ZSI6IHRydWUsICJtb2RpZnki0iB0cnVLL
CAicmVtb3ZlIjogdHJlZlZlLCAiX2ljb24i0iAibWdtdC5ob3N0X2luZm8ifSwgeyJpbmZvIjogdHJlZSwgIm5hbWU
i0iAiY2NfbWfUyWdLciIsICJ0aXRzZSI6ICJYwxsLWNlbnRlciIsICJjcmVhdGU0i0iB0cnVLLCAibW9kaWZ5Ijogd
HJlZSwgInJlbW92ZSI6IHRydWUsICJjaGlsZHJlbiI6Ift7ImLuZm8i0iB0cnVLLCAibGVhZiI6IHRydWUsICJuYW1
lIjogImNjX2FnZW50X2luZm8iLCAidG0bGU0i0iAiqWd0bnQ0LCAiY3JlYXRlIjogdHJlZSwgIm1vZGhmeSI6IHRyd
WUsICJyZW1vdmUi0iB0cnVllfSwgeyJpbmZvIjogdHJlZSwgImxlyWYi0iB0cnVLLCAibmFtZSI6ICJyY19xdWVlZV9
pbmZvIiwgInRpdGxLIjogIlF1ZXVlIiwgImNyZWf0ZSI6IHRydWUsICJtb2RpZnki0iB0cnVLLCAicmVtb3ZlIjogd
HJlZlZlLCAiX2ljb24i0iAiY2MubWfUyWdLciJ9LCB7ImLuZm8i0iB0cnVLLCAibGVhZiI6IHRydWUsICJuYW1lIjo
gInN1YnNjcmliZXJfcG9ydGFsIiwgInRpdGxLIjogIlN1YnNjcmliZXIgcG9ydGFsIiwgImNyZWf0ZSI6IHRydWUsI
CJtb2RpZnki0iB0cnVLLCAicmVtb3ZlIjogdHJlZSwgIl9pY29uIjogInN1YnNjcmliZXIucG9ydGFsIn0sIHsiaW5
mbyI6IHRydWUsICJsZWf0ZSI6IHRydWUsICJfaWNvbiI6ICJtc3IuZHJhdyJ
9LCB7ImLuZm8i0iB0cnVLLCAibGVhZiI6IHRydWUsICJuYW1lIjogImRvbWfPpbL90Y3BfZHVtcCIsICJ0aXRzZSI6I
CJQQ0FQIHRyYWNlIG1hbmFnZXIiLCAiY3JlYXRlIjogdHJlZSwgIm1vZGhmeSI6IHRydWUsICJyZW1vdmUi0iB0cnV
LLCAiX2ljb24i0iAiZG9tYWluLnRjcF9kdW1wIn0sIHsiaW5mbyI6IHRydWUsICJsZWf0ZSI6IHRydWUsICJyZW1vdmUi0
iAizG9tYWluX3RlbGVjb25mIiwgInRpdGxLIjogIlRlbGVjb25mZXJlbmNlIG1hbmFnZXIiLCAiY3JlYXRlIjogdHJ
lZSwgIm1vZGhmeSI6IHRydWUsICJyZW1vdmUi0iB0cnVLLCAiX2ljb24i0iAiZG9tYWluLnRlbGVjb25mIn0sIHsia
W5mbyI6IHRydWUsICJsZWf0ZSI6IHRydWUsICJyZW1vdmUi0iB0cnVLLCAiX2ljb24i0iAiZG9tYWluX251bWJlcl9wbGFuIiwgInRpdGxLIjogIk5
1bWJlcmLuZyBwbGFuIG1hbmFnZXIiLCAiY3JlYXRlIjogdHJlZSwgIm1vZGhmeSI6IHRydWUsICJyZW1vdmUi0iB0c
nVLLCAiX2ljb24i0iAiZG9tYWluLm51bWJlcl9wbGFuIn0sIHsiaW5mbyI6IHRydWUsICJsZWf0ZSI6IHRydWUsICJtb
2RpZnki0iB0cnVLLCAicmVtb3ZlIjogdHJlZSwgIl9pY29uIjogInN0YXRpc3RpY3MuZG9tYWluIn0sIHsiaW5mbyI
6IHRydWUsICJsZWf0ZSI6IHRydWUsICJyZW1vdmUi0iAibXNyX21lZGhIiwigInRpdGxLIjogIk1TUiBtZWRpYSBtYW5hZ
2VyIiwgImNyZWf0ZSI6IHRydWUsICJtb2RpZnki0iB0cnVLLCAicmVtb3ZlIjogdHJlZSwgIl9pY29uIjogIm1zci5
tZWRpYSJ9LCB7ImLuZm8i0iB0cnVLLCAibmFtZSI6ICJkb21haW5fY2FsZW5kYXIiLCAidG0bGU0i0iAiQ2FsZW5kY
XIiLCAiY3JlYXRlIjogdHJlZSwgIm1vZGhmeSI6IHRydWUsICJyZW1vdmUi0iB0cnVLLCAiY2hpbGRyZW4i0iBbeyJ
pbmZvIjogdHJlZSwgImxlyWYi0iB0cnVLLCAibmFtZSI6ICJzaW1wbGVfc2V0dGluZ3NfQ2FsZW5kYXIiLCAidG0b
GU0i0iAiQ2FsZW5kYXIiLCAiY3JlYXRlIjogdHJlZSwgIm1vZGhmeSI6IHRydWUsICJyZW1vdmUi0iB0cnVllfSwgeyJ
pbmZvIjogdHJlZSwgImxlyWYi0iB0cnVLLCAibmFtZSI6ICJTaGVkdWxLIiwgInRpdGxLIjogIlRpbWV0YWJsZXMiL
CAiY3JlYXRlIjogdHJlZSwgIm1vZGhmeSI6IHRydWUsICJyZW1vdmUi0iB0cnVllfV0sICJfaWNvbiI6ICJkb21haW4
uY2FsZW5kYXIifSwgeyJpbmZvIjogdHJlZSwgImxlyWYi0iB0cnVLLCAibmFtZSI6ICJtc3JfcmlvbnRpdGxLIiwgInRpdGxLIjogIk1TUiBtZWRpYSByZWdpc3RyYXJzIiwgImNyZWf0ZSI6IHRydWUsICJtb2RpZnki0iB0cnVLLCAicmV
tb3ZlIjogdHJlZSwgIl9pY29uIjogIm1zci5yZWdpc3RyYXIIifv19fQ=="/>
</role>
</result>
</out>

```

hc_role_properties_set

Команда изменяет и(или) добавляет свойство для существующей CoCon роли через HTTP-терминал.

Аналог команды в CoCon:

`cocon/role/.properties/set`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/role_properties_set

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set specified properties to exists role.
      Usage: http://server:9999/commands/role\_properties\_set.
      Implementation: hc\_role\_properties\_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- propertyType -->
  <xs:complexType name="propertyTypeIn">
    <xs:attribute type="xs:string" name="key" use="required"/>
    <xs:attribute type="xs:string" name="value" use="required"/>
  </xs:complexType>

  <!-- roleTypeIn -->
  <xs:complexType name="roleTypeIn">
    <xs:sequence>
      <xs:element name="property" type="propertyTypeIn" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="role" type="roleTypeIn" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- propertyType -->
  <xs:complexType name="propertyTypeOut">
    <xs:attribute type="xs:string" name="key" use="optional"/>

```



```

    <xs:attribute type="xs:string" name="value" use="optional"/>
  </xs:complexType>

  <!-- roleTypeOut -->
  <xs:complexType name="roleTypeOut">
    <xs:sequence>
      <xs:element name="property" type="propertyTypeOut" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:element name="role" type="roleTypeOut" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="result" type="resultType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/role_properties_set

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_role_properties_set.xsd">
  <request>
    <role name="rolename3">
      <property key="one" value="value"/>
      <property key="two" value="value1\nvalue2"/>
    </role>
    <role name="admin1_role">
      <property key="one" value="value1"/>
    </role>
  </request>
</in>

```

Ответ:

```
status: 201  
content:
```

[Hc_role_remove](#)

Команда предназначена для удаления CoCon роли через HTTP-терминал.

Аналог команды в CoCon:

`cocon/role/remove`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/role_remove`

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Remove exists role with specific name.
      Usage: http://server:9999/commands/role_remove.
      Implementation: hc_role_remove.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- roleType -->
  <xs:complexType name="roleType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="role" type="roleType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/role_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_role_remove.xsd">
  <request>
    <role name="rolename3" />
  </request>
</in>
```

Ответ:

```
status: 201
content:
```

[Hc_set_role_to_user](#)

Команда назначает CoCon роль пользователю через HTTP-терминал.

Аналог команды в CoCon:

cocon/set-role-from-user

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/set_role_to_user

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set exists role to exists user.
      Usage: http://server:9999/commands/set_role_to_user.
      Implementation: hc_set_role_to_user.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- userType -->
  <xs:complexType name="userType">
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- roleType -->
  <xs:complexType name="roleType">
    <xs:sequence>
      <xs:element name="user" type="userType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="role" type="roleType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>

```

```

    <xs:choice>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/set_role_to_user

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_set_role_to_user.xsd">
  <request>
    <role name="oper_role">
      <user name="oper"/>
    </role>
  </request>
</in>

```

Ответ:

```

status: 201
content:

```

[Hc_del_role_from_user](#)

Команда удаляет существующую CoCon роль существующего пользователя через HTTP-терминал.

Аналог команды в CoCon:

cocon/del-role-from-user

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/del_role_from_user

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:annotation>
    <xs:documentation xml:lang="en">
      Delete role from exists user.
      Usage: http://server:9999/commands/del_role_from_user.
      Implementation: hc_del_role_from_user.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- userType -->
  <xs:complexType name="userType">
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- roleType -->
  <xs:complexType name="roleType">
    <xs:sequence>
      <xs:element name="user" type="userType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute type="xs:string" name="name" use="required"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="role" type="roleType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>

```

```

    <xs:choice>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/del_role_from_user

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_del_role_from_user.xsd">
  <request>
    <role name="oper_role">
      <user name="oper"/>
    </role>
  </request>
</in>

```

Ответ:

```

status: 201
content:

```

Hc_routing - Управление маршрутизацией

- [Hc_change_context](#)
- [Hc_list_of_context](#)
- [Hc_show_context](#)
- [Hc_delete_context](#)

Hc_change_context

Команда предназначена для импорта контекста в домен (Виртуальную АТС). Если контекст с данным именем уже существует, он будет обновлен, если контекст маршрутизации с данным именем не существует, он будет создан.

Аналог команды в CoCon:

```
/domain/<DOMAIN>/routing/import
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/change_context

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />
  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- progressType -->
  <xs:complexType name="progressType">
    <xs:sequence>
      <xs:element name="ref" minOccurs="1" maxOccurs="1" type="valueType" />
      <xs:element name="text" minOccurs="1" maxOccurs="1" type="valueType" />
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="context" type="xs:base64Binary" />
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required" />
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="error" type="errorType" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- progress -->
  <xs:element name="progress" type="progressType"/>
</xs:schema>

```

❗ В base64 должен быть закодирован целевой контекст маршрутизации, то есть XML-элемент типа contextType (тип определен в context.xsd).

context.xsd:

Схема XML-файла определена в разделе **show_context**.

Пример:

Запрос: http://192.168.1.21:9999/commands/change_context

Параметры:

```
<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request domain="ct.office">

<context>PGNvbnRleHQgeG1sbnM6eHM9Imh0dHA6Ly93d3cudzMub3JnLzIwMDEvWE1MU2NoZW1hLWluc3RhbmNlIiB4czpu
b05hbWVzcGFjZVNjaGVtYUxvY2F0aW9uPSJlY3NzX3JvdXRpbmcueHNkIgotbmFtZT0iY3R4X3RvX2V3c2QiIG5wPSJucF9vZ
mZpY2UiIGRpZ210bWFWPSJhdXRvIj4KICA8cnVsZSBuYW1lPSJ0b19ld3NkX3Nvcml0PjogICAgPGNvbmlRpdGlvbnM+CiAgIC
AgIDxjZHBuIGRpZ210cz0iJSIvPgogICAgPC9jb25kaXRpb25zPgogICAgPGFjdGlvbnM+CiAgICAgIDxjZHBuIGRpZ210cz0
ieyV9Ii8+CiAgICA8L2FjdGlvbnM+CiAgICA8cmVzdWx0PgogICAgICA8ZXh0ZXJlYyYw+CiAgICAgICAgPHRydW5rIHZhbHVl
PSJzbWctNCIgd2VpZ2h0PSIxIi8+CiAgICAgIDwvZXh0ZXJlYyYw+CiAgICA8L3Jlc3VsD4KICA8L3J1bGU+CjwvY29udGV4d
D4K</context>
  </request>
</in>
```

Данная base64 строка соответствует контексту:

```
<context xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ecss_routing.xsd"
name="ctx_to_ewsd" np="np_office" digitmap="auto">
  <rule name="to_ewsd_sorm">
    <conditions>
      <cdpn digits="%"/>
    </conditions>
    <actions>
      <cdpn digits="{%}"/>
    </actions>
    <result>
      <external>
        <trunk value="smg-4" weight="1"/>
      </external>
    </result>
  </rule>
</context>
```

[Hc_list_of_context](#)

Аналог команды в CoCon отсутствует, просмотреть доступные в системе контексты маршрутизации можно, например, введя команду `domain/<DOMAIN>/routing/show` и нажать `tab`.

Команда возвращает список контекстов маршрутизации импортированных в домен (Виртуальную АТС).

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/list_of_context

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

200 – в случае успеха;

404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- contextType-->
  <xs:complexType name="contextType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="contexts">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="context" type="contextType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/list_of_context

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="list_of_context.xsd">
  <request domain="biysk.local"/>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="list_of_context.xsd">
  <contexts>
    <context name="ctx_from_ewsd"/>
    <context name="default_routing"/>
    <context name="ctx_to_ivr"/>
    <context name="ctx_city_common"/>
    <context name="ctx_to_ewsd"/>
    <context name="ctx_from_local"/>
  </contexts>
</out>
```

hc_show_context

Аналог команд в CoCon:

```
/cluster/storage/<CLUSTER>/routing/show test.domain
/domain/<DOMAIN>/routing/show
```

Команда позволяет просмотреть содержание контекста маршрутизации.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/show_context

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

200 – в случае успеха;

404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа::

hc_show_context.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="context" type="xs:base64Binary"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="context" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

ecss_routing.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- type for calling arguments-->
  <xs:simpleType name="atomType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
  <xs:simpleType name="booleanType">
    <xs:restriction base="xs:boolean"/>
  </xs:simpleType>
  <xs:simpleType name="termType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
  <xs:simpleType name="listType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
  <xs:simpleType name="binaryType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
  <xs:simpleType name="integerType">
    <xs:restriction base="xs:integer"/>
  </xs:simpleType>
  <xs:simpleType name="positiveIntegerType">
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="0"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="negativeIntegerType">
    <xs:restriction base="xs:integer">
      <xs:maxInclusive value="0"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="floatType">
    <xs:restriction base="xs:float"/>
  </xs:simpleType>
  <xs:simpleType name="positiveFloatType">
    <xs:restriction base="xs:float">
      <xs:minInclusive value="0"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="negativeFloatType">
    <xs:restriction base="xs:float">
      <xs:maxInclusive value="0"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="stringType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>

```



```

<xs:simpleType name="nonemptyStringType">
  <xs:restriction base="xs:string"/>
</xs:simpleType>
<xs:simpleType name="enumType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="first"/>
    <xs:enumeration value="second"/>
    <xs:enumeration value="third"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="digitmapType">
  <xs:restriction base="xs:string"/>
</xs:simpleType>
<xs:simpleType name="enumDaysType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="sun"/>
    <xs:enumeration value="mon"/>
    <xs:enumeration value="thu"/>
  </xs:restriction>
</xs:simpleType>
<!-- externalRoutingType -->
<xs:complexType name="externalRoutingType">
  <xs:attribute name="service" type="xs:string" use="required"/>
  <xs:attribute name="id" type="xs:string" use="optional">
    <xs:annotation>
      <xs:documentation>
        Service specific identifier. Used to distribute different rules by different
service's instances.
        For radius_route_service id - the RADIUS servers' ids separated by semicolon
(;)
      </xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="args" type="xs:string" use="optional"/>
  <xs:attribute name="timeout" type="positiveIntegerType" use="optional"/>
</xs:complexType>
<xs:complexType name="contextType">
  <xs:sequence>
    <xs:element name="rule" maxOccurs="unbounded">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="conditions" minOccurs="0">
            <xs:complexType>
              <xs:all>
                <xs:element name="calling" type="conditionCallingType"
minOccurs="0"/>
                <xs:element name="called" type="calledType" minOccurs="0"/>
                <xs:element name="rgn" type="rgnConditionType" minOccurs="0"/>
              </xs:all>
            </xs:complexType>
          </xs:element>
          <xs:element name="cdpn" type="cdpnType" minOccurs="0"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

```

```

        <xs:element name="cgpn" type="cgpnType" minOccurs="0"/>
        <xs:element name="ocdpn" type="ocdpnConditionType"
minOccurs="0"/>

        <xs:element name="time" type="valueType" minOccurs="0"/>
        <xs:element name="date" type="valueType" minOccurs="0"/>
        <xs:element name="weekday" type="weekdayType" minOccurs="0"/>
        <xs:element name="timetable" type="valueType" minOccurs="0"/>
        <xs:element name="tag" type="valueType" minOccurs="0"/>
        <xs:element name="final" type="boolType" minOccurs="0"/>
        <xs:element name="cause" type="causeType" minOccurs="0"/>
    </xs:all>
</xs:complexType>
</xs:element>
<xs:element name="actions" minOccurs="0">
    <xs:complexType>
        <xs:all>
            <xs:element name="external_routing"
type="externalRoutingType" minOccurs="0"/>
            <xs:element name="set_options" type="optionsType"
minOccurs="0"/>

            <xs:element name="cgpn" type="cgpnType" minOccurs="0"/>
            <xs:element name="cdpn" type="cdpnType" minOccurs="0"/>
            <xs:element name="restore_cgpn" type="nullType" minOccurs="0"
/>

            <xs:element name="restore_cdpn" type="nullType" minOccurs="0"
/>

            <xs:element name="restore_rgn" type="nullType" minOccurs="0"/>

            <xs:element name="empty_rgn" type="nullType" minOccurs="0"/>
            <xs:element name="empty_ocdpn" type="nullType" minOccurs="0"/>

            <xs:element name="restore_ocdpn" type="nullType"
minOccurs="0"/>

            <xs:element name="calling" type="actionCallingType"
minOccurs="0"/>

            <xs:element name="called" type="calledActionType"
minOccurs="0"/>

            <xs:element name="rgn" type="rgnActionType" minOccurs="0"/>
            <xs:element name="ocdpn" type="ocdpnActionType" minOccurs="0"
/>

            <xs:element name="callerid" type="xs:string" minOccurs="0"/>
            <xs:element name="final" type="boolType" minOccurs="0"/>
            <xs:element name="cause" type="causeType" minOccurs="0"/>
            <xs:element name="alarm" type="alarmType" minOccurs="0"/>
            <xs:element name="log" type="logType" minOccurs="0"/>
        </xs:all>
    </xs:complexType>
</xs:element>
<xs:element name="result">
    <xs:complexType>

```

```

        <xs:choice>
          <xs:element name="local" type="localResultType"/>
          <xs:element name="teleconference"
type="teleconferenceResultType"/>
          <xs:element name="ivr" type="ivrResultType"/>
          <xs:element name="direction" type="directionResultType"/>
          <xs:element name="external" type="externalResultType"/>
          <xs:element name="incomplete" type="incompleteResultType"/>
          <xs:element name="no_route" type="no_routeResultType"/>
          <xs:element name="continue" type="continueResultType"/>
          <xs:element name="next" type="nextResultType"/>
        </xs:choice>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="description" type="xs:string" use="optional"/>
</xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="name" type="xs:string" use="required">
  <xs:annotation>
    <xs:documentation>The name of the current context.</xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="domain" type="xs:string" use="optional">
  <xs:annotation>
    <xs:documentation>Deprecated. Now domain name get form the context of the
commands' execution.</xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="type" type="contextTypeType" use="optional">
  <xs:annotation>
    <xs:documentation>Deprecated. Now domain name get form the context of the
commands' execution.</xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="np" type="xs:string" default="default" use="optional">
  <xs:annotation>
    <xs:documentation>Numbering plan for current routing context. By default np =
default.</xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="digitmap" type="digitmapType" use="optional">
  <xs:annotation>
    <xs:documentation>Manual-set digitmap for the current routing context.</
xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="description" type="xs:string" use="optional">

```

```

        <xs:annotation>
            <xs:documentation>Text description of the current routing context.</
xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <!--obsolete-->
</xs:complexType>
<xs:element name="context" type="contextType"/>
<!-- actionCallingType -->
<xs:complexType name="actionCallingType">
    <xs:attribute name="category" type="categoryType" use="optional"/>
    <xs:attribute name="caller_id" type="stringType" use="optional"/>
    <xs:attribute name="display_name" type="stringType" use="optional"/>
    <xs:attribute name="sorm_digits" type="stringType" use="optional"/>
    <xs:attribute name="sorm_ni" type="enumNiType" use="optional"/>
</xs:complexType>
<!-- conditionCallingType -->
<xs:complexType name="conditionCallingType">
    <xs:attribute name="access_private" type="booleanType" use="optional"/>
    <xs:attribute name="access_local" type="booleanType" use="optional"/>
    <xs:attribute name="access_zone" type="booleanType" use="optional"/>
    <xs:attribute name="access_international" type="booleanType" use="optional"/>
    <xs:attribute name="access_intercity" type="booleanType" use="optional"/>
    <xs:attribute name="access_emergency" type="booleanType" use="optional"/>
    <xs:attribute name="have_access_to" type="atomType" use="optional"/>
    <xs:attribute name="category" type="categoryType" use="optional"/>
    <xs:attribute name="caller_id" type="stringType" use="optional"/>
    <xs:attribute name="display_name" type="stringType" use="optional"/>
    <xs:attribute name="sorm_digits" type="stringType" use="optional"/>
    <xs:attribute name="sorm_ni" type="enumNiType" use="optional"/>
    <xs:attribute name="interface_group" type="stringType" use="optional"/>
    <xs:attribute name="iface" type="binaryType" use="optional"/>
    <xs:attribute name="city" type="stringType" use="optional"/>
    <xs:attribute name="region" type="stringType" use="optional"/>
    <xs:attribute name="operator" type="stringType" use="optional"/>
</xs:complexType>
<xs:complexType name="calledType">
    <xs:attribute name="sorm_digits" type="stringType" use="optional"/>
    <xs:attribute name="sorm_ni" type="enumNiType" use="optional"/>
    <xs:attribute name="city" type="stringType" use="optional"/>
    <xs:attribute name="region" type="stringType" use="optional"/>
    <xs:attribute name="operator" type="stringType" use="optional"/>
</xs:complexType>
<xs:complexType name="calledActionType">
    <xs:attribute name="sorm_digits" type="stringType" use="optional"/>
    <xs:attribute name="sorm_ni" type="enumNiType" use="optional"/>
</xs:complexType>
<xs:simpleType name="contextTypeType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="local"/>
    </xs:restriction>
</xs:simpleType>

```

```

        <xs:enumeration value="transit"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="categoryType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="unknownAtThisTime"/>
        <xs:enumeration value="operatorFrench"/>
        <xs:enumeration value="operatorEnglish">
            <xs:annotation>
                <xs:documentation xml:lang="en">Obsolete value. Correct value is
operatorEnglish</xs:documentation>
            </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="operatorEnglish"/>
        <xs:enumeration value="operatorGerman"/>
        <xs:enumeration value="operatorRussian"/>
        <xs:enumeration value="operatorSpanish"/>
        <xs:enumeration value="reserved"/>
        <xs:enumeration value="ordinarySubscriber"/>
        <xs:enumeration value="subscriberWithPriority"/>
        <xs:enumeration value="dataCall"/>
        <xs:enumeration value="testCall"/>
        <xs:enumeration value="spare"/>
        <xs:enumeration value="payphone"/>
        <xs:enumeration value="category0"/>
        <xs:enumeration value="hotelsSubscriber"/>
        <xs:enumeration value="freeSubscriber"/>
        <xs:enumeration value="paidSubscriber"/>
        <xs:enumeration value="localSubscriber"/>
        <xs:enumeration value="localTaksofon"/>
        <xs:enumeration value="autoCallI"/>
        <xs:enumeration value="semiautoCallI"/>
        <xs:enumeration value="autoCallII"/>
        <xs:enumeration value="semiautoCallII"/>
        <xs:enumeration value="autoCallIII"/>
        <xs:enumeration value="semiautoCallIII"/>
        <xs:enumeration value="autoCallIV"/>
        <xs:enumeration value="semiautoCallIV"/>
        <xs:enumeration value="0"/>
        <xs:enumeration value="1"/>
        <xs:enumeration value="2"/>
        <xs:enumeration value="3"/>
        <xs:enumeration value="4"/>
        <xs:enumeration value="5"/>
        <xs:enumeration value="6"/>
        <xs:enumeration value="7"/>
        <xs:enumeration value="8"/>
        <xs:enumeration value="9"/>
        <xs:enumeration value="10"/>
        <xs:enumeration value="11"/>
    </xs:restriction>
</xs:simpleType>

```

```
<xs:enumeration value="12"/>
<xs:enumeration value="13"/>
<xs:enumeration value="14"/>
<xs:enumeration value="15"/>
<xs:enumeration value="16"/>
<xs:enumeration value="17"/>
<xs:enumeration value="18"/>
<xs:enumeration value="19"/>
<xs:enumeration value="20"/>
<xs:enumeration value="21"/>
<xs:enumeration value="22"/>
<xs:enumeration value="23"/>
<xs:enumeration value="24"/>
<xs:enumeration value="25"/>
<xs:enumeration value="26"/>
<xs:enumeration value="27"/>
<xs:enumeration value="28"/>
<xs:enumeration value="29"/>
<xs:enumeration value="30"/>
<xs:enumeration value="31"/>
<xs:enumeration value="32"/>
<xs:enumeration value="33"/>
<xs:enumeration value="34"/>
<xs:enumeration value="35"/>
<xs:enumeration value="36"/>
<xs:enumeration value="37"/>
<xs:enumeration value="38"/>
<xs:enumeration value="39"/>
<xs:enumeration value="40"/>
<xs:enumeration value="41"/>
<xs:enumeration value="42"/>
<xs:enumeration value="43"/>
<xs:enumeration value="44"/>
<xs:enumeration value="45"/>
<xs:enumeration value="46"/>
<xs:enumeration value="47"/>
<xs:enumeration value="48"/>
<xs:enumeration value="49"/>
<xs:enumeration value="50"/>
<xs:enumeration value="51"/>
<xs:enumeration value="52"/>
<xs:enumeration value="53"/>
<xs:enumeration value="54"/>
<xs:enumeration value="55"/>
<xs:enumeration value="56"/>
<xs:enumeration value="57"/>
<xs:enumeration value="58"/>
<xs:enumeration value="59"/>
<xs:enumeration value="60"/>
<xs:enumeration value="61"/>
```

```
<xs:enumeration value="62"/>
<xs:enumeration value="63"/>
<xs:enumeration value="64"/>
<xs:enumeration value="65"/>
<xs:enumeration value="66"/>
<xs:enumeration value="67"/>
<xs:enumeration value="68"/>
<xs:enumeration value="69"/>
<xs:enumeration value="70"/>
<xs:enumeration value="71"/>
<xs:enumeration value="72"/>
<xs:enumeration value="73"/>
<xs:enumeration value="74"/>
<xs:enumeration value="75"/>
<xs:enumeration value="76"/>
<xs:enumeration value="77"/>
<xs:enumeration value="78"/>
<xs:enumeration value="79"/>
<xs:enumeration value="80"/>
<xs:enumeration value="81"/>
<xs:enumeration value="82"/>
<xs:enumeration value="83"/>
<xs:enumeration value="84"/>
<xs:enumeration value="85"/>
<xs:enumeration value="86"/>
<xs:enumeration value="87"/>
<xs:enumeration value="88"/>
<xs:enumeration value="89"/>
<xs:enumeration value="90"/>
<xs:enumeration value="91"/>
<xs:enumeration value="92"/>
<xs:enumeration value="93"/>
<xs:enumeration value="94"/>
<xs:enumeration value="95"/>
<xs:enumeration value="96"/>
<xs:enumeration value="97"/>
<xs:enumeration value="98"/>
<xs:enumeration value="99"/>
<xs:enumeration value="100"/>
<xs:enumeration value="101"/>
<xs:enumeration value="102"/>
<xs:enumeration value="103"/>
<xs:enumeration value="104"/>
<xs:enumeration value="105"/>
<xs:enumeration value="106"/>
<xs:enumeration value="107"/>
<xs:enumeration value="108"/>
<xs:enumeration value="109"/>
<xs:enumeration value="110"/>
<xs:enumeration value="111"/>
```

```
<xs:enumeration value="112"/>
<xs:enumeration value="113"/>
<xs:enumeration value="114"/>
<xs:enumeration value="115"/>
<xs:enumeration value="116"/>
<xs:enumeration value="117"/>
<xs:enumeration value="118"/>
<xs:enumeration value="119"/>
<xs:enumeration value="120"/>
<xs:enumeration value="121"/>
<xs:enumeration value="122"/>
<xs:enumeration value="123"/>
<xs:enumeration value="124"/>
<xs:enumeration value="125"/>
<xs:enumeration value="126"/>
<xs:enumeration value="127"/>
<xs:enumeration value="128"/>
<xs:enumeration value="129"/>
<xs:enumeration value="130"/>
<xs:enumeration value="131"/>
<xs:enumeration value="132"/>
<xs:enumeration value="133"/>
<xs:enumeration value="134"/>
<xs:enumeration value="135"/>
<xs:enumeration value="136"/>
<xs:enumeration value="137"/>
<xs:enumeration value="138"/>
<xs:enumeration value="139"/>
<xs:enumeration value="140"/>
<xs:enumeration value="141"/>
<xs:enumeration value="142"/>
<xs:enumeration value="143"/>
<xs:enumeration value="144"/>
<xs:enumeration value="145"/>
<xs:enumeration value="146"/>
<xs:enumeration value="147"/>
<xs:enumeration value="148"/>
<xs:enumeration value="149"/>
<xs:enumeration value="150"/>
<xs:enumeration value="151"/>
<xs:enumeration value="152"/>
<xs:enumeration value="153"/>
<xs:enumeration value="154"/>
<xs:enumeration value="155"/>
<xs:enumeration value="156"/>
<xs:enumeration value="157"/>
<xs:enumeration value="158"/>
<xs:enumeration value="159"/>
<xs:enumeration value="160"/>
<xs:enumeration value="161"/>
```



```
<xs:enumeration value="162"/>
<xs:enumeration value="163"/>
<xs:enumeration value="164"/>
<xs:enumeration value="165"/>
<xs:enumeration value="166"/>
<xs:enumeration value="167"/>
<xs:enumeration value="168"/>
<xs:enumeration value="169"/>
<xs:enumeration value="170"/>
<xs:enumeration value="171"/>
<xs:enumeration value="172"/>
<xs:enumeration value="173"/>
<xs:enumeration value="174"/>
<xs:enumeration value="175"/>
<xs:enumeration value="176"/>
<xs:enumeration value="177"/>
<xs:enumeration value="178"/>
<xs:enumeration value="179"/>
<xs:enumeration value="180"/>
<xs:enumeration value="181"/>
<xs:enumeration value="182"/>
<xs:enumeration value="183"/>
<xs:enumeration value="184"/>
<xs:enumeration value="185"/>
<xs:enumeration value="186"/>
<xs:enumeration value="187"/>
<xs:enumeration value="188"/>
<xs:enumeration value="189"/>
<xs:enumeration value="190"/>
<xs:enumeration value="191"/>
<xs:enumeration value="192"/>
<xs:enumeration value="193"/>
<xs:enumeration value="194"/>
<xs:enumeration value="195"/>
<xs:enumeration value="196"/>
<xs:enumeration value="197"/>
<xs:enumeration value="198"/>
<xs:enumeration value="199"/>
<xs:enumeration value="200"/>
<xs:enumeration value="201"/>
<xs:enumeration value="202"/>
<xs:enumeration value="203"/>
<xs:enumeration value="204"/>
<xs:enumeration value="205"/>
<xs:enumeration value="206"/>
<xs:enumeration value="207"/>
<xs:enumeration value="208"/>
<xs:enumeration value="209"/>
<xs:enumeration value="210"/>
<xs:enumeration value="211"/>
```

```
<xs:enumeration value="212"/>
<xs:enumeration value="213"/>
<xs:enumeration value="214"/>
<xs:enumeration value="215"/>
<xs:enumeration value="216"/>
<xs:enumeration value="217"/>
<xs:enumeration value="218"/>
<xs:enumeration value="219"/>
<xs:enumeration value="220"/>
<xs:enumeration value="221"/>
<xs:enumeration value="222"/>
<xs:enumeration value="223"/>
<xs:enumeration value="224"/>
<xs:enumeration value="225"/>
<xs:enumeration value="226"/>
<xs:enumeration value="227"/>
<xs:enumeration value="228"/>
<xs:enumeration value="229"/>
<xs:enumeration value="230"/>
<xs:enumeration value="231"/>
<xs:enumeration value="232"/>
<xs:enumeration value="233"/>
<xs:enumeration value="234"/>
<xs:enumeration value="235"/>
<xs:enumeration value="236"/>
<xs:enumeration value="237"/>
<xs:enumeration value="238"/>
<xs:enumeration value="239"/>
<xs:enumeration value="240"/>
<xs:enumeration value="241"/>
<xs:enumeration value="242"/>
<xs:enumeration value="243"/>
<xs:enumeration value="244"/>
<xs:enumeration value="245"/>
<xs:enumeration value="246"/>
<xs:enumeration value="247"/>
<xs:enumeration value="248"/>
<xs:enumeration value="249"/>
<xs:enumeration value="250"/>
<xs:enumeration value="251"/>
<xs:enumeration value="252"/>
<xs:enumeration value="253"/>
<xs:enumeration value="254"/>
<xs:enumeration value="255"/>
</xs:restriction>
</xs:simpleType>
<xs:complexType name="cdpnType">
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
</xs:complexType>
</xs:schema>
```

```

    <xs:attribute name="inni" type="inniType"/>
    <xs:attribute name="npi" type="npiType"/>
    <xs:attribute name="ni" type="enumNiType"/>
    <xs:attribute name="in_list" type="xs:string"/>
</xs:complexType>
<xs:complexType name="rgnConditionType">
    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="nai" type="naiType"/>
    <xs:attribute name="incomplete" type="xs:boolean"/>
    <xs:attribute name="apri" type="apriType"/>
    <xs:attribute name="npi" type="npiType"/>
    <xs:attribute name="ni" type="enumNiType"/>
    <xs:attribute name="in_list" type="xs:string"/>
    <xs:attribute name="empty" type="booleanType">
        <xs:annotation>
            <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>
<xs:complexType name="ocdpnConditionType">
    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="nai" type="naiType"/>
    <xs:attribute name="ni" type="enumNiType"/>
    <xs:attribute name="npi" type="npiType"/>
    <xs:attribute name="apri" type="apriType"/>
    <xs:attribute name="category" type="categoryType"/>
    <xs:attribute name="incomplete" type="xs:boolean"/>
    <xs:attribute name="in_list" type="xs:string"/>
    <xs:attribute name="empty" type="booleanType">
        <xs:annotation>
            <xs:documentation>In case of empty attribute set - other attributes MUST NOT be
set.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>
<xs:complexType name="ocdpnActionType">
    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="nai" type="naiType"/>
    <xs:attribute name="ni" type="enumNiType"/>
    <xs:attribute name="npi" type="npiType"/>
    <xs:attribute name="apri" type="apriType"/>
    <xs:attribute name="category" type="categoryType"/>
    <xs:attribute name="incomplete" type="xs:boolean"/>
</xs:complexType>
<xs:complexType name="rgnActionType">
    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="nai" type="naiType"/>
    <xs:attribute name="incomplete" type="xs:boolean"/>
    <xs:attribute name="apri" type="apriType"/>

```

```

    <xs:attribute name="npi" type="npiType"/>
    <xs:attribute name="ni" type="enumNiType"/>
</xs:complexType>
<xs:simpleType name="naiType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="spare"/>
    <xs:enumeration value="subscriberNumber"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="nationalNumber"/>
    <xs:enumeration value="internationalNumber"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="inniType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="routingToInternalNumberAllowed"/>
    <xs:enumeration value="routingToInternalNumberNotAllowed"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="npiType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="spare"/>
    <xs:enumeration value="isdnTelephony"/>
    <xs:enumeration value="dataNumberingPlan"/>
    <xs:enumeration value="telexNumberingPlan"/>
    <xs:enumeration value="reserved1"/>
    <xs:enumeration value="reserved2"/>
    <xs:enumeration value="reserved3"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="enumNiType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="private"/>
    <xs:enumeration value="local"/>
    <xs:enumeration value="zone"/>
    <xs:enumeration value="intercity"/>
    <xs:enumeration value="international"/>
    <xs:enumeration value="emergency"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="cgpnType">
  <xs:attribute name="nai" type="naiType"/>
  <xs:attribute name="incomplete" type="xs:boolean"/>
  <xs:attribute name="npi" type="npiType"/>
  <xs:attribute name="apri" type="apriType"/>
  <xs:attribute name="screening" type="screeningType"/>
  <xs:attribute name="digits" type="xs:string"/>
  <xs:attribute name="ni" type="enumNiType"/>
  <xs:attribute name="in_list" type="xs:string"/>
</xs:complexType>
<xs:simpleType name="apriType">

```

```

    <xs:restriction base="xs:string">
      <xs:enumeration value="presentationAllowed"/>
      <xs:enumeration value="presentationRestricted"/>
      <xs:enumeration value="addressNotAvailable"/>
      <xs:enumeration value="spare"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="screeningType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="userProvidedNotVerified"/>
      <xs:enumeration value="userProvidedVerifiedAndPassed"/>
      <xs:enumeration value="userProvidedVerifiedAndFailed"/>
      <xs:enumeration value="networkProvided"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="weekdayType">
    <xs:attribute name="value" type="xs:string" use="optional"/>
    <xs:attribute name="day_types" type="xs:string" use="optional"/>
  </xs:complexType>
  <xs:simpleType name="maxLoadType">
    <xs:restriction base="xs:string">
      <xs:pattern value="([0-9]+)|([0-9]{1,2}%)|(100%)/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="trunkType">
    <xs:attribute name="value" type="xs:string" use="required"/>
    <xs:attribute name="weight" type="xs:integer" use="optional"/>
    <xs:attribute name="max_load" type="maxLoadType" use="optional"/>
  </xs:complexType>
  <xs:complexType name="boolType">
    <xs:attribute name="value" type="xs:boolean" use="required"/>
  </xs:complexType>
  <xs:complexType name="alarmType">
    <xs:attribute name="severity" type="alarmSeverity" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="logType">
    <xs:attribute name="severity" type="logSeverity" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="causeType">
    <xs:attribute name="value" type="xs:string" use="optional">
      <xs:annotation xml:lang="en">
        <xs:documentation>Obsolete attribute. Now are used acp attribute.</
xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>

```

```

    <xs:attribute name="acp" type="xs:string" use="optional">
      <xs:annotation xml:lang="en">
        <xs:documentation>The list of the ACP cause separated by comma or space.</
xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="sip" type="xs:string" use="optional">
      <xs:annotation xml:lang="en">
        <xs:documentation>The list of the SIP cause separated by comma or space.</
xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="isup" type="xs:string" use="optional">
      <xs:annotation xml:lang="en">
        <xs:documentation>The list of the ISUP cause separated by comma or space.</
xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <xs:simpleType name="alarmSeverity">
    <xs:restriction base="xs:string">
      <xs:enumeration value="warning"/>
      <xs:enumeration value="minor"/>
      <xs:enumeration value="major"/>
      <xs:enumeration value="critical"/>
      <xs:enumeration value="indeterminate"/>
      <xs:enumeration value="cleared"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="logSeverity">
    <xs:restriction base="xs:string">
      <xs:enumeration value="error"/>
      <xs:enumeration value="warning"/>
      <xs:enumeration value="info"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="nullType"/>
  <xs:complexType name="optionsType">
    <xs:sequence>
      <xs:element name="option" type="keyvalType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="keyvalType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="externalResultType">
    <xs:sequence>
      <xs:element name="trunk" type="trunkType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

```

```

</xs:complexType>
<xs:complexType name="templateExternalResultType">
  <xs:sequence>
    <xs:element name="trunk" type="trunkType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="template_trunk" type="trunkType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!-- Result types -->
<xs:complexType name="localResultType">
  <xs:choice>
    <xs:element name="continue" type="continueResultType" minOccurs="0" maxOccurs="1"/>
  </xs:choice>
  <xs:attribute name="vdn" type="xs:string" use="optional"/>
</xs:complexType>
<xs:complexType name="nextResultType">
  <xs:attribute name="tag" type="xs:string" use="optional"/>
</xs:complexType>
<xs:complexType name="callcenterResultType"/>
<xs:complexType name="teleconferenceResultType"/>
<xs:complexType name="ivrResultType">
  <xs:attribute name="script" type="xs:string" use="required"/>
</xs:complexType>
<xs:complexType name="directionResultType">
  <xs:attribute name="value" type="xs:string" use="required"/>
</xs:complexType>
<xs:complexType name="incompleteResultType">
  <xs:attribute name="timeout" type="xs:decimal" use="optional"/>
</xs:complexType>
<xs:complexType name="no_routeResultType">
  <xs:attribute name="isup_cause" type="xs:string" use="optional"/>
</xs:complexType>
<xs:complexType name="continueResultType">
  <xs:attribute name="tag" type="xs:string" use="optional"/>
  <xs:attribute name="context" type="xs:string" use="optional"/>
</xs:complexType>
</xs:schema>

```

Более подробную информацию о параметрах можно посмотреть в разделе: [Детальное описание конфигурационного файла маршрутизации](#) .

Пример:

Запрос: http://192.168.1.21:9999/commands/show_context

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_show_context.xsd">
  <request domain="biysk.local" context="ctx_city_common"/>
</in>

```


hc_delete_context

Команда предназначена для удаления контекста маршрутизации из системы.

Аналог команды в CoCon:

`/domain/test.domain/routing/delete`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/delete_context`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

204 – в случае успеха;

404 – в случае ошибки.

XSD-схема XML-файла с запросом:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="context" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/delete_context

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_delete_context.xsd">
  <request domain="ct.office" context="default_routing" />
</in>
```

Ответ: 204

Нс_security_profile - команды управления профилями безопасности

- [Нс_security_profile_clean](#)
- [Нс_security_profile_info](#)
- [Нс_security_profile_set](#)

[Нс_security_profile_clean](#)

Команда позволяет вернуть одно или все свойства профиля безопасности к значениям по умолчанию

Аналог команд в CoCon:

/system/security/profile/clean

/domain/<DOMAIN>/security/profile/clean

Шаблон URL-запроса:

http://host:port/commands/nc_security_profile_clean

Коды ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

Xsd схема XML запроса/ответа

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="profileType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="optionType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="0"
maxOccurs="1"/>
              <xs:element name="profile" type="profileType"/>
              <xs:element name="option" type="optionType" minOccurs="0"
maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!--Positive case-->
        <xs:element name="response" fixed="ok"/>
        <!--Or error-->
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/security_profile_clean

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_security_profile_clean.xsd">
  <request>
    <profile name="test.profile"/>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_security_profile_clean.xsd">
  <response>ok</response>
</out>
```

[Hc_security_profile_info](#)

Команда позволяет просмотреть список и свойства профилей безопасности

Аналог команд в CoCon:

/system/security/profile/info

/domain/<DOMAIN>/security/profile/info

Шаблон URL-запроса:

http://host:port/commands/hc_security_profile_info

Коды ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

Xsd схема XML запроса/ответа

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="profileType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="profileOutType">
    <xs:choice>
      <xs:element name="password_restrictions">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="password_restriction" type="restrictionType"
minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="password_restrictions" fixed="default"/>
    </xs:choice>
    <xs:attribute name="system" type="xs:string" use="required"/>
    <xs:attribute name="show_password_at_cli" type="xs:string" use="required"/>
    <xs:attribute name="show_password_at_api" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="restrictionType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="args" type="xs:string" use="required"/>
    <xs:attribute name="msg" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="0"
maxOccurs="1"/>
              <xs:element name="profile" type="profileType" minOccurs="0"
maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

        </xs:all>
    </xs:complexType>
</xs:element>

<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <!--Positive case-->
            <xs:element name="response">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="profile" type="profileOutType" minOccurs="0"
maxOccurs="unbounded"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <!--Or error-->
            <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/security_profile_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_security_profile_info.xsd">
  <request>
    <domain name="d.city"/>
    <profile name="callcenter"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_security_profile_info.xsd">
  <response>
    <profile system="callcenter" show_password_at_cli="true" show_password_at_api="true">
      <password_restrictions/>
    </profile>
  </response>
</out>
```

hc_security_profile_set

Команда позволяет изменить список профилей безопасности

Аналог команд в CoSop:

/system/security/profile/set

/domain/<DOMAIN>/security/profile/set

Шаблон URL-запроса:

http://host:port/commands/hc_security_profile_set

Коды ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

Xsd схема XML запроса/ответа

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="profilesType">
    <xs:sequence>
      <xs:element name="profile" type="profileType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="profileType">
    <xs:choice>
      <xs:element name="password_restrictions">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="password_restriction" type="restrictionType"
minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="password_restrictions" fixed="default"/>
    </xs:choice>
    <xs:attribute name="system" type="xs:string" use="required"/>
    <xs:attribute name="show_password_at_cli" type="xs:string" use="required"/>
    <xs:attribute name="show_password_at_api" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="restrictionType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="args" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" type="domainType" minOccurs="0"/>
              <xs:element name="profiles" type="profilesType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```



```

    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <!--Positive case-->
        <xs:element name="response" fixed="ok"/>
        <!--Or error-->
        <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/security_profile_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_security_profile_set.xsd">
  <request>
    <profiles>
      <profile system="test_profile" show_password_at_cli="true"
show_password_at_api="default">
        <password_restrictions>default</password_restrictions>
      </profile>
    </profiles>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_security_profile_set.xsd">
  <response>ok</response>
</out>

```

Нс_sip_cluster_network - Управление сетевыми настройками SIP адаптера в кластере

- [Нс_sip_cluster_network_clean](#)
- [Нс_sip_cluster_network_info](#)
- [Нс_sip_cluster_network_set](#)

[Нс_sip_cluster_network_clean](#)

Команда позволяет восстановить в значение по умолчанию сетевые настройки SIP адаптера на уровне кластера.

Аналог команды в CoCon:

```
cluster/adapter/<SIP_ADAPTER>/sip/network/clean
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_cluster_network_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean network properties of SIP cluster.
      Usage: http://server:9999/commands/sip\_cluster\_network\_clean.
      Implementation: hc_sip_general_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="cluster_options" type="clusterCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="cluster_options" type="clusterCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример**Запрос:**

http://192.168.1.21:9999/commands/sip_cluster_network_clean

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sip_cluster_network_clean.xsd">
  <request>
    <cluster_options>
      <cluster name="sip1">
        <option name="ip_set"/>
      </cluster>
    </cluster_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sip_cluster_network_clean.xsd">
  <result>
    <cluster_options>
      <cluster name="sip1">
        <success option="ip_set"/>
      </cluster>
    </cluster_options>
  </result>
</out>

```

Нс_sip_cluster_network_info

Команда позволяет получить сетевые настройки SIP адаптера на уровне кластера.

Аналог команды в CoCon:

cluster/adapter/<SIP_ADAPTER>/sip/network/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_cluster_network_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show network settings of SIP cluster.
      Usage: http://server:9999/commands/sip\_cluster\_network\_info.
      Implementation: hc_sip_cluster_network_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- sipClusterNetworkOptionsType -->
  <xs:complexType name="sipClusterNetworkOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="ip_set" type="stringType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Format:
  

IpSetName_1:Port_11,Port_12, ...:DSCP_1:Node_11#Host_11,Node_12#Host_12, ...;IpSetName_2:Port_21,Port_22, ...:DSCP_2:Node_21#Host_21,Node_22#Host_22, ...;...
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="cluster_options" type="clusterInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="cluster_options" type="clusterInfoInType" minOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример**Запрос:**

http://192.168.1.21:9999/commands/sip_cluster_network_info

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="sip_cluster_network_info.xsd">
    <request>
        <cluster_options cluster="sip1" />
    </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sip_cluster_network_info.xsd">
    <result>
        <cluster_options>
            <cluster name="sip1">
                <options xs:type="sipClusterNetworkOptionsType" ip_set="ipset1:5060,5061,5062,5063,5064,5065,5070,5071,5072,5073,5074,5075:24:sip1@ecss1#192.168.2.61,sip1@ecss2#192.168.2.62;ipset2:5090,5091:0:sip1@ecss1#192.168.2.61,sip1@ecss2#127.0.0.1"/>
            </cluster>
        </cluster_options>
    </result>
</out>

```

Ис sip_cluster_network_set

Команда позволяет изменить сетевые настройки SIP адаптера на уровне кластера.

Аналог команды в CoCon:

`cluster/adapter/<SIP_ADAPTER>/sip/network/set`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/sip_cluster_network_set`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set network properties of SIP cluster.
      Usage: http://server:9999/commands/sip\_cluster\_network\_set.
      Implementation: hc_sip_general_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- sipClusterNetworkOptionsType -->
  <xs:complexType name="sipClusterNetworkOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="ip_set" type="stringType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Format:
              IpSetName_1:Port_11,Port_12, ...:DSCP_1:Node_11#Host_11,Node_12#Host_12, ...;IpSetName_2:Port_21,Port_22, ...:DSCP_2:Node_21#Host_21,Node_22#Host_22, ...;...
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="cluster_options" type="clusterSetOutType"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="cluster_options" type="clusterSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```



```

    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/sip_cluster_network_set

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sip_cluster_network_set.xsd">
  <request>
    <cluster_options>
      <cluster name="sip1">
        <options xs:type="sipClusterNetworkOptionsType" ip_set="ipset1:5060,5061,5062,5063,5064,5065,5071,5072,5073,5074,5075:sip1@ecss1#192.168.2.61,sip1@ecss2#192.168.2.62;ipset2:5080,5081,5082:24:sip1@ecss1#192.168.2.61,sip1@ecss2#192.168.2.62;"/>
      </cluster>
    </cluster_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sip_cluster_network_set.xsd">
  <result>
    <cluster_options>
      <cluster name="sip1">
        <success option="ip_set"/>
      </cluster>
    </cluster_options>
  </result>
</out>
```

Hc_sip_domain_network - Управление сетевыми настройками протокола SIP на уровне домена

- [Hc_sip_domain_network_info](#)
- [Hc_sip_domain_network_clean](#)
- [Hc_sip_domain_network_set](#)

[Hc_sip_domain_network_info](#)

Команда позволяет просмотреть сетевые настройки протокола SIP на уровне домена.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/network/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_domain_network_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show sip-domain properties of SIP cluster.
      Usage: http://server:9999/commands/sip_domain_network_info.
      Implementation: hc_sip_domain_network_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- sipDomainPropertiesOptionsType-->
  <xs:complexType name="sipDomainNetworkOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="ip_sets" type="stringType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Usage: ip_set1,ip_set_2,...,ip_set_N
              The list of IP set's separated by comma.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Внимание: Параметр *Domain* является обязательным для данной команды.

Пример

Запрос:

http://192.168.1.21:9999/commands/sip_domain_network_info

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="sip_domain_network_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_domain_network_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="sipDomainNetworkOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="sipDomainNetworkOptionsType" ip_sets="ipset1"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Http_sip_domain_network_clean

Команда позволяет восстановить сетевые настройки SIP адаптера на уровне домена (доступно только администратору софтсвича)

Аналог команды в CoCon:

`domain/<DOMAIN>/sip/network/clean`

Метод http запроса:

POST

Шаблон http запроса:

`http://host:port/commands/sip_domain_network_clean`

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean network properties of SIP cluster.
      Usage: http://server:9999/commands/sip_cluster_network_clean.
      Implementation: hc_sip_general_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="cluster_options" type="clusterCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="cluster_options" type="clusterCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_domain_network_clean

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_domain_network_clean.xsd">
  <request>
    <domain_options>
      <system/>
      <domain name="ct.office">
        <option name="ip_set"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_domain_network_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="ct.office">
        <success option="ip_set"/>
      </domain>
    </domain_options>
  </result>
</out>

```

[Hc_sip_domain_network_set](#)

Команда предназначена для изменения сетевых настроек SIP адаптера на уровне домена (доступно только администратору софтсвича)

Аналог команды в CoCon:

domain/<DOMAIN>/sip/network/set

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/sip_domain_network_set

Код http ответа:

200 – в случае успеха;
404 – в случае ошибки.

Xsd схема XML запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set SIP domain's network settings.
      Usage: http://server:9999/commands/sip_domain_network_set.
      Implementation: hc_sip_domain_network_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2017, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- sipDomainNetworkOptionsType-->
  <xs:complexType name="sipDomainNetworkOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="ip_set" type="stringType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Format: IpSet_1,IpSet_2; ...
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->

```

```

<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/sip_domain_network_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_domain_network_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="sipDomainNetworkOptionsType" />
      </system>
      <domain name="biysk.local">
        <options xs:type="sipDomainNetworkOptionsType" ip_set="ipset1" />
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_domain_network_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="ct.office">
        <success option="ip_set"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Hc_sip_domain_properties - Управление свойствами SIP на уровне виртуальной АТС

- [Hc_sip_domain_properties_clean](#)
- [Hc_sip_domain_properties_info](#)
- [Hc_sip_domain_properties_set](#)

[Hc_sip_domain_properties_clean](#)

Команда устанавливает значение по умолчанию для свойств SIP на уровне виртуальной АТС.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/properties/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_domain_properties_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean sip-domain properties of SIP cluster.
      Usage: http://server:9999/commands/sip_domain_properties_clean.
      Implementation: hc_sip_domain_properties_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Параметр "Domain" является обязательным для данной команды.

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_domain_properties_clean

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sip_domain_properties_clean.xsd">
  <request>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <option name="mtu_size"/>
        <option name="sip_transport"/>
      </domain>
      <domain name="d.empty"/>
    </domain_options>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_domain_properties_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="mtu_size"/>
        <success option="sip_transport"/>
      </domain>
    </domain_options>
  </result>
</out>
```

[Hc_sip_domain_properties_info](#)

Команда позволяет просмотреть свойства SIP на уровне домена.

Аналог команды в CoCon:**domain/<DOMAIN>/sip/properties/info****Метод HTTP-запроса:**

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_domain_properties_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show sip-domain properties of SIP cluster.
      Usage: http://server:9999/commands/sip\_domain\_properties\_info.
      Implementation: hc_sip_domain_properties_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- compactHeadersType -->
  <xs:simpleType name="compactHeadersType">
    <xs:restriction base="unionType">
      <xs:enumeration value="force_compact"/>
      <xs:enumeration value="force_long"/>
      <xs:enumeration value="default"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- authenticationStoreType -->
  <xs:simpleType name="authenticationStoreType">
    <xs:restriction base="unionType">
      <xs:enumeration value="ldap"/>
      <xs:enumeration value="ds"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- sipTransportType -->
  <xs:simpleType name="sipTransportType">
    <xs:restriction base="unionType">
      <xs:enumeration value="udp_only"/>
      <xs:enumeration value="udp_prefer"/>
      <xs:enumeration value="tcp_prefer"/>
      <xs:enumeration value="tcp_only"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- sipDomainPropertiesOptionsType-->
  <xs:complexType name="sipDomainPropertiesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="remote_party_id_enable" type="booleanType" use="optional"/>
        <xs:attribute name="direct_early_update" type="booleanType" use="optional"/>
        <xs:attribute name="absence_allow_by_rfc" type="booleanType" use="optional"/>
        <xs:attribute name="disable_ruri_control" type="booleanType" use="optional"/>
        <xs:attribute name="non_telephone_number_allow" type="booleanType" use="optional"
/>
        <xs:attribute name="compact_headers" type="compactHeadersType" use="optional"/>

```

```

<xs:attribute name="silent_mode" type="booleanType" use="optional"/>
<xs:attribute name="info_header" type="booleanType" use="optional"/>
<xs:attribute name="via_use_to_check" type="booleanType" use="optional"/>
<xs:attribute name="nit100" type="booleanType" use="optional"/>
<xs:attribute name="realm" type="stringType" use="optional"/>
<xs:attribute name="allowed_ruri_to" type="stringType" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Format: Ruri_1; ruri_2; ...
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="auto_answer_version" type="stringType" use="optional"/>
<xs:attribute name="authentication_store" type="authenticationStoreType" use="opt
ional"/>

<xs:attribute name="sip_transport" type="sipTransportType" use="optional"/>
<xs:attribute name="mtu_size" type="positiveIntegerType" use="optional"/>
<xs:attribute name="dtmf_relay" type="booleanType" use="optional"/>
<xs:attribute name="dtmf_duration" type="positiveIntegerType" use="optional"/>
<xs:attribute name="symbol_hash_as_is" type="booleanType" use="optional"/>
<xs:attribute name="category_to_sip" type="stringType" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Defaults value:
        "category"
        "cpc"
        "cpc-rus"
        "default"
        "none"
      In addition to the values given,
      any other may be specified.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="messages_localization_enable" type="booleanType" use="optiona
l"/>

  </xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">

```



```

        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoInType" minOccurs="
1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

 Параметр "Domain" является обязательным для данной команды.

Пример:

Запрос: http://192.168.1.21:9999/commands/sip_domain_properties_info

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="sip_domain_properties_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_domain_properties_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="sipDomainPropertiesOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="sipDomainPropertiesOptionsType" via_use_to_check="false"
          symbol_hash_as_is="false" sip_transport="udp_only" silent_mode="false"
          remote_party_id_enable="false" realm="biysk.local" non_telephone_number_allow="false"
          nit100="false" mtu_size="1300" messages_localization_enable="false" info_header="true"
          dtmf_relay="false" dtmf_duration="100" disable_ruri_control="false" direct_early_update="false"
          compact_headers="default" category_to_sip="category" auto_answer_version="alert-info:Auto
          Answer" authentication_store="ds" absence_allow_by_rfc="false"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_sip_domain_properties_set

Команда позволяет редактировать свойства SIP на уровне домена.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/properties/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_domain_properties_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set sip-domain-properties properties of SIP cluster.
      Usage: http://server:9999/commands/sip_domain_properties_set.
      Implementation: hc_sip_domain_properties_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- compactHeadersType -->
  <xs:simpleType name="compactHeadersType">
    <xs:restriction base="unionType">
      <xs:enumeration value="force_compact"/>
      <xs:enumeration value="force_long"/>
      <xs:enumeration value="default"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- authenticationStoreType -->
  <xs:simpleType name="authenticationStoreType">
    <xs:restriction base="unionType">
      <xs:enumeration value="ldap"/>
      <xs:enumeration value="ds"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- sipTransportType -->
  <xs:simpleType name="sipTransportType">
    <xs:restriction base="unionType">
      <xs:enumeration value="udp_only"/>
      <xs:enumeration value="udp_prefer"/>
      <xs:enumeration value="tcp_prefer"/>
      <xs:enumeration value="tcp_only"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- sipDomainPropertiesOptionsType-->
  <xs:complexType name="sipDomainPropertiesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="remote_party_id_enable" type="booleanType" use="optional"/>
        <xs:attribute name="direct_early_update" type="booleanType" use="optional"/>
        <xs:attribute name="absence_allow_by_rfc" type="booleanType" use="optional"/>
        <xs:attribute name="disable_ruri_control" type="booleanType" use="optional"/>
        <xs:attribute name="non_telephone_number_allow" type="booleanType" use="optional"
/>
        <xs:attribute name="compact_headers" type="compactHeadersType" use="optional"/>

```

```

<xs:attribute name="silent_mode" type="booleanType" use="optional"/>
<xs:attribute name="info_header" type="booleanType" use="optional"/>
<xs:attribute name="via_use_to_check" type="booleanType" use="optional"/>
<xs:attribute name="nit100" type="booleanType" use="optional"/>
<xs:attribute name="realm" type="stringType" use="optional"/>
<xs:attribute name="allowed_ruri_to" type="stringType" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Format: Ruri_1; ruri_2; ...
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="auto_answer_version" type="stringType" use="optional"/>
<xs:attribute name="authentication_store" type="authenticationStoreType"
use="optional"/>
<xs:attribute name="sip_transport" type="sipTransportType" use="optional"/>
<xs:attribute name="mtu_size" type="positiveIntegerType" use="optional"/>
<xs:attribute name="dtmf_relay" type="booleanType" use="optional"/>
<xs:attribute name="dtmf_duration" type="positiveIntegerType" use="optional"/>
<xs:attribute name="symbol_hash_as_is" type="booleanType" use="optional"/>
<xs:attribute name="category_to_sip" type="stringType" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Defaults value:
        "category"
        "cpc"
        "cpc-rus"
        "default"
        "none"
      In addition to the values given,
      any other may be specified.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="messages_localization_enable" type="booleanType"
use="optional"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">

```

```

        <xs:complexType>
            <xs:all>
                <xs:element name="domain_options" type="domainSetInType"/>
            </xs:all>
        </xs:complexType>
    </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
            <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

 Параметр "Domain" является обязательным для данной команды.

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_domain_properties_set

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request>
    <domain_options>
      <system>
        <options xs:type="sipDomainPropertiesOptionsType"/>
      </system>
      <domain name="ct.office">
        <options xs:type="sipDomainPropertiesOptionsType" compact_headers="force_long"
mtu_size="1400"/>
      </domain>
      <domain name="biysk.local">
        <options xs:type="sipDomainPropertiesOptionsType" sip_transport="udp_only"
mtu_size="1300" realm="biysk.local"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_domain_properties_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="ct.office">
        <success option="compact_headers"/>
        <success option="mtu_size"/>
      </domain>
      <domain name="biysk.local">
        <success option="mtu_size"/>
        <success option="realm"/>
        <success option="sip_transport"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Нс_sip_dynamic_user_area - Команды управления динамически конфигурируемыми интерфейсами

- [Нс_sip_dynamic_user_area_info](#)
- [Нс_sip_dynamic_user_area_set](#)

Нс_sip_dynamic_user_area_info

Команда выводит информацию об одной или нескольких динамических группах SIP-абонентов.

Аналог команды в CoSoc:

domain/<DOMAIN>/sip/user/dynamic-area/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_dynamic_user_area_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <!-- request type -->
  <xs:complexType name="requestType">
    <xs:attribute name="area" use="optional"/>
    <xs:attribute name="domain" use="required"/>
  </xs:complexType>

  <!-- answer type -->
  <xs:complexType name="areaType">
    <xs:attribute name="error" use="optional"/>
    <xs:attribute name="servers" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          STATUS
          The service provides information about the server in the form of ID and
          Sample: "id:master,status:ok;id:slave,status:unknown"
        </xs:documentation>
        <xs:documentation xml:lang="ru">
          Сервис предоставляет информацию о сервере в виде ID и STATUS
          Пример: "id:master,status:ok;id:slave,status:unknown"
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="context" use="optional"/>
    <xs:attribute name="sip_domain" use="optional"/>
    <xs:attribute name="size_limit" use="optional"/>
    <xs:attribute name="full_digest" use="optional"/>
    <xs:attribute name="radius_draft" use="optional"/>
    <xs:attribute name="modification_rule" use="optional"/>
    <xs:attribute name="cisco_vsa_for_all" use="optional"/>
  </xs:complexType>

  <!-- response type -->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="area" type="areaType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

</xs:element>

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="response" type="responseType" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример 1:

http://192.168.1.21:9999/commands/sip_dynamic_user_area_info

```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sip_dynamic_user_area_info.xsd">
  <request domain="biysk.local"/>
</in>

```

Ответ:

```

status: 200
content:
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_sip_dyn
amic_user_area_info.xsd">
  <response>
    <area sip_domain="dyn_bsk2" context="ctx_from_local" size_limit="10" full_digest="true"
radius_draft="false" cisco_vsa_for_all="false" servers="id:r1,status:ok;id:r2,status:ok"/>
    <area sip_domain="make_dyn" context="ctx_from_local" size_limit="1" full_digest="true"
radius_draft="false" cisco_vsa_for_all="false" servers="id:r1,status:ok;id:r2,status:ok"/>
  </response>
</out>

```

Пример 2:

http://192.168.1.21:9999/commands/sip_dynamic_user_area_info


```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sip_dynamic_user_area_info.xsd">
  <request domain="biysk.local" area="superarea"/>
</in>

```

Ответ:

```

status: 200
content:
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_sip_dyn
amic_user_area_info.xsd">
  <response>
    <area sip_domain="superarea" error="No information about this areas:
    &quot;superarea&quot;"/>
  </response>
</out>

```

hc_sip_dynamic_user_area_set

Команда используется для установки параметров группы динамических SIP-абонентов.

- 1 - Не существовавшие - создаются;
- 2 - Существующие - изменяются;
- 3 - Не указанные - удаляются;
- 4 - Дубликаты - не допустимы.

Аналог команды в CoSop:

domain/<DOMAIN>/sip/user/dynamic-area/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_dynamic_user_area_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- request type -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="area" type="areaType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" use="required"/>
  </xs:complexType>

  <!-- area type -->
  <xs:complexType name="areaType">
    <xs:attribute name="servers" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Service in accordance with several server names in the form of a list.
          Sample: "id:master;id:slave"
        </xs:documentation>
        <xs:documentation xml:lang="ru">
          Сервис в качестве настроек принимает лишь названия серверов в виде списка.
          Пример: "id:master;id:slave"
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="context" use="required"/>
    <xs:attribute name="size_limit" use="required"/>
    <xs:attribute name="sip_domain" use="optional"/>
    <xs:attribute name="full_digest" use="optional"/>
    <xs:attribute name="modification_rule" use="optional"/>
    <xs:attribute name="cisco_vsa_for_all" use="optional"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>

```

```

        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1:

Запрос: http://192.168.23.34:9999/commands/sip_dynamic_user_area_set

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_sip_dynamic_user_area_set.xsd">
    <request domain="ivan.local">
        <area sip_domain="make" context="default_routing" size_limit="1" full_digest="true"
modification_rule="no" cisco_vsa_for_all="false" servers="id:asdasdqa,status:unknown"/>
        <area sip_domain="change" context="default_routing" size_limit="1" full_digest="true"
modification_rule="no" cisco_vsa_for_all="false" servers="id:asdasdqa,status:unknown"/>
    </request>
</in>

```

Ответ:

```

status: 204
content:

```

Пример 2:

Запрос: http://192.168.23.34:9999/commands/sip_dynamic_user_area_set

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="hc_sip_dynamic_user_area_set.xsd">
    <request domain="ivan.local">
        <area sip_domain="make" context="default_routing" size_limit="1" full_digest="true"
modification_rule="no" cisco_vsa_for_all="false" servers="id:asdasdqa,status:unknown"/>
        <area sip_domain="make" context="default_routing" size_limit="1" full_digest="true"
modification_rule="no" cisco_vsa_for_all="false" servers="id:asdasdqa,status:unknown"/>
        <area sip_domain="make" context="default_routing" size_limit="1" full_digest="true"
modification_rule="no" cisco_vsa_for_all="false" servers="id:asdasdqa,status:unknown"/>
    </request>
</in>

```

Ответ:

```

status: 200
content:

<?xml version="1.0"?>
  <out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_sip_dynamic_user_area_set.xsd">
    <error cmd="sip_dynamic_user_area_set" reason="duplicate" body="make;make"/>
  </out>

```

Пример 3:

Запрос: http://192.168.1.21:9999/commands/sip_dynamic_user_area_set

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_sip_dynamic_user_area_set.xsd">
  <request domain="biysk.local">
    <area sip_domain="make_dyn" context="ctx_from_local" size_limit="1" full_digest="true"
modification_rule="no" cisco_vsa_for_all="false" servers="id:r1;id:r2"/>
    <area sip_domain="dyn_bsk2" context="ctx_from_local" size_limit="10" full_digest="true"
modification_rule="no" cisco_vsa_for_all="false" servers="id:r1;id:r2;id:blabla"/>
  </request>
</in>

```

Ответ:

```

response
status: 204
content:

```

Нс_sip_error_messages - управление текстовыми описаниями ошибок SIP

- [Нс_sip_error_messages_clean](#)
- [Нс_sip_error_messages_info](#)
- [Нс_sip_error_messages_set](#)

[Нс_sip_error_messages_clean](#)

Команда возвращает дефолтные значения текстовых описаний для ошибок SIP

Аналог команды в CoCon:

domain/<DOMAIN>/sip/sip-error-messages/clean

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/sip-error-messages_clean

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/sip_error_messages_clean

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_sip_error_messages_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="code_1"/>
        <option name="external"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sip_error_messages_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="code_1"/>
        <success option="external"/>
      </domain>
    </domain_options>
  </result>
</out>
```

[Hc_sip_error_messages_info](#)

Команда возвращает текстовые описания для SIP-cause

Аналог команды в CoCon:

domain/<DOMAIN>/sip/sip-error-messages/info

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/sip-error-messages_info

Код http ответа

200 – в случае успеха;
404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="sipErrorMessagesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="code_0" type="xs:string" use="optional" />
        <xs:attribute name="code_1" type="xs:string" use="optional" />
        <xs:attribute name="code_2" type="xs:string" use="optional" />
        <xs:attribute name="code_3" type="xs:string" use="optional" />
        <xs:attribute name="code_4" type="xs:string" use="optional" />
        <xs:attribute name="code_5" type="xs:string" use="optional" />
        <xs:attribute name="code_6" type="xs:string" use="optional" />
        <xs:attribute name="code_7" type="xs:string" use="optional" />
        <xs:attribute name="code_8" type="xs:string" use="optional" />
        <xs:attribute name="code_9" type="xs:string" use="optional" />
        <xs:attribute name="code_10" type="xs:string" use="optional" />
        <xs:attribute name="code_11" type="xs:string" use="optional" />
        <xs:attribute name="code_12" type="xs:string" use="optional" />
        <xs:attribute name="code_14" type="xs:string" use="optional" />
        <xs:attribute name="code_15" type="xs:string" use="optional" />
        <xs:attribute name="code_16" type="xs:string" use="optional" />
        <xs:attribute name="network_not_configured" type="xs:string" use="optional" />
        <xs:attribute name="alias_declaration_error" type="xs:string" use="optional" />
        <xs:attribute name="invalid_alias_digits" type="xs:string" use="optional" />
        <xs:attribute name="wrong_identifier" type="xs:string" use="optional" />
        <xs:attribute name="bad_template" type="xs:string" use="optional" />
        <xs:attribute name="name_is_busy" type="xs:string" use="optional" />
        <xs:attribute name="unknown_password" type="xs:string" use="optional" />
        <xs:attribute name="unknown_login" type="xs:string" use="optional" />
        <xs:attribute name="no_ldap_connection" type="xs:string" use="optional" />
        <xs:attribute name="execution_error" type="xs:string" use="optional" />
        <xs:attribute name="licence_limit" type="xs:string" use="optional" />
        <xs:attribute name="bad_value" type="xs:string" use="optional" />
        <xs:attribute name="wrong_set" type="xs:string" use="optional" />
        <xs:attribute name="cancelled" type="xs:string" use="optional" />
        <xs:attribute name="error_length" type="xs:string"
use="optional" />
        <xs:attribute name="error_contact_already_declared" type="xs:string"
use="optional" />
        <xs:attribute name="error_channel_id_not_found" type="xs:string"
use="optional" />
        <xs:attribute name="external" type="xs:string" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```



```

</xs:complexType>

<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/sip_error_messages_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_sip_error_messages_info.xsd">
  <request>
    <domain_options domain="biysk.local" />
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sip_error_messages_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="sipErrorMessagesOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="sipErrorMessagesOptionsType"
wrong_set="language:default,text:Wrong parameters to set;language:en,text:Wrong parameters to
set;language:ru,text:Неверный набор параметров" wrong_identifier="language:default,text:Wrong
identifier;language:en,text:Wrong identifier;language:ru,text:Неверный идентификатор"
unknown_password="language:default,text:password must be specified for
login;language:en,text:password must be specified for login;language:ru,text:Для входа требуется
указать пароль" unknown_login="language:default,text:login must be specified before
password;language:en,text:login must be specified before password;language:ru,text:Логин должен
быть указан перед паролем" no_ldap_connection="language:default,text:No connection with LDAP
server;language:en,text:No connection with LDAP server;language:ru,text:Нет соединения с LDAP
сервером" network_not_configured="language:default,text:Network parameters (node_ip or share_set)
must be configured before;language:en,text:Network parameters (node_ip or share_set) must be
configured before;language:ru,text:" name_is_busy="language:default,text:Identifiers name is
busy;language:en,text:Identifiers name is busy;language:ru,text:Идентификатор занят"
licence_limit="language:default,text:Licence limit is exceeded!;language:en,text:Licence limit
is exceeded!;language:ru,text:Превышение лицензионных ограничений"
invalid_alias_digits="language:default,text:Invalid digits in alias
number;language:en,text:Invalid digits in alias number;language:ru,text:" external=""
execution_error="language:default,text:Unexpected execution error;language:en,text:Unexpected
execution error;language:ru,text:Неожиданная ошибка выполнения"
error_length="language:default,text:Different length of ranges;language:en,text:Different length
of ranges;language:ru,text:Разная длина диапазонов"
error_contact_already_declared="language:default,text:This contact already
mapped;language:en,text:This contact already mapped;language:ru,text:Контакт уже сопоставлен"
error_channel_id_not_found="language:default,text:This channel id not
declared;language:en,text:This channel id not declared;language:ru,text:Идентификатор канала не
объявлен" code_9="language:default,text:Wrong authentication;language:en,text:Wrong
authentication;language:ru,text:Ошибка аутентификации" code_8="language:default,text:Resource
limit;language:en,text:Resource limit;language:ru,text:Ограничение ресурсов"
code_7="language:default,text:Request URI to disallowed address;language:en,text:Request URI to
disallowed address;language:ru,text:" code_6="language:default,text:Public call isn't
allowed;language:en,text:Public call isn't allowed;language:ru,text:Внешние вызовы запрещены"
code_5="language:default,text:Internal error at process of
registration;language:en,text:Internal error at process of
registration;language:ru,text:Внутренняя ошибка регистрации"
code_4="language:default,text:Request from disallowed IP;language:en,text:Request from disallowed
IP;language:ru,text:Запрос с запрещенного IP" code_3="language:default,text:The registration
domain is wrong or missed;language:en,text:The registration domain is wrong or
missed;language:ru,text:Неверный домен регистрации" code_2="language:default,text:The user name
is wrong or missed;language:en,text:The user name is wrong or missed;language:ru,text:Неверный

```

```

логин" code_16="language:default,text:Wrong parameters set;language:en,text:Wrong parameters
set;language:ru,text:Неправильный набор параметров" code_15="language:default,text:Anonymity
Disallowed;language:en,text:Anonymity Disallowed;language:ru,text:"
code_14="language:default,text:Interface is blocked;language:en,text:Interface is
blocked;language:ru,text:Интерфейс заблокирован" code_12="language:default,text:Contacts list is
empty;language:en,text:Contacts list is empty;language:ru,text:Пустой список контактов"
code_11="language:default,text:Calls are disallowed;language:en,text:Calls are
disallowed;language:ru,text:Вызовы запрещены" code_10="language:default,text:Third party
registration is disallowed;language:en,text:Third party registration is
disallowed;language:ru,text:" code_1="language:default,text:Unknown subscriber (User name or/and
registration domain are wrong);language:en,text:Unknown subscriber (User name or/and registration
domain are wrong);language:ru,text:Неизвестный подписчик (неверный логин и/или домен
регистрации)" code_0="language:default,text:Internal system error;language:en,text:Internal
system error;language:ru,text:Внутренняя ошибка системы"
cancelled="language:default,text:Operation was cancelled;language:en,text:Operation was
cancelled;language:ru,text:Операция отменена" bad_value="language:default,text:Wrong (unknown or
unspecified) value;language:en,text:Wrong (unknown or unspecified)
value;language:ru,text:Неверное/неизвестное значение" bad_template="language:default,text:Bad
templates definition;language:en,text:Bad templates definition;language:ru,text:"
alias_declaration_error="language:default,text:Aliases declaration
error;language:en,text:Aliases declaration error;language:ru,text:"/>
    </domain>
  </domain_options>
</result>
</out>

```

Нс_sip_error_messages_set

Команда задает значения текстовых описаний для SIP-cause

Аналог команды в CoCon:

domain/<DOMAIN>/sip/sip-error-messages/set

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/sip_error_messages_set

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="sipErrorMessagesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="code_0" type="xs:string" use="optional" />
        <xs:attribute name="code_1" type="xs:string" use="optional" />
        <xs:attribute name="code_2" type="xs:string" use="optional" />
        <xs:attribute name="code_3" type="xs:string" use="optional" />
        <xs:attribute name="code_4" type="xs:string" use="optional" />
        <xs:attribute name="code_5" type="xs:string" use="optional" />
        <xs:attribute name="code_6" type="xs:string" use="optional" />
        <xs:attribute name="code_7" type="xs:string" use="optional" />
        <xs:attribute name="code_8" type="xs:string" use="optional" />
        <xs:attribute name="code_9" type="xs:string" use="optional" />
        <xs:attribute name="code_10" type="xs:string" use="optional" />
        <xs:attribute name="code_11" type="xs:string" use="optional" />
        <xs:attribute name="code_12" type="xs:string" use="optional" />
        <xs:attribute name="code_14" type="xs:string" use="optional" />
        <xs:attribute name="code_15" type="xs:string" use="optional" />
        <xs:attribute name="code_16" type="xs:string" use="optional" />
        <xs:attribute name="network_not_configured" type="xs:string" use="optional" />
        <xs:attribute name="alias_declaration_error" type="xs:string" use="optional" />
        <xs:attribute name="invalid_alias_digits" type="xs:string" use="optional" />
        <xs:attribute name="wrong_identififier" type="xs:string" use="optional" />
        <xs:attribute name="bad_template" type="xs:string" use="optional" />
        <xs:attribute name="name_is_busy" type="xs:string" use="optional" />
        <xs:attribute name="unknown_password" type="xs:string" use="optional" />
        <xs:attribute name="unknown_login" type="xs:string" use="optional" />
        <xs:attribute name="no_ldap_connection" type="xs:string" use="optional" />
        <xs:attribute name="execution_error" type="xs:string" use="optional" />
        <xs:attribute name="licence_limit" type="xs:string" use="optional" />
        <xs:attribute name="bad_value" type="xs:string" use="optional" />
        <xs:attribute name="wrong_set" type="xs:string" use="optional" />
        <xs:attribute name="cancelled" type="xs:string" use="optional" />
        <xs:attribute name="error_length" type="xs:string"
use="optional" />
        <xs:attribute name="error_contact_already_declared" type="xs:string"
use="optional" />
        <xs:attribute name="error_channel_id_not_found" type="xs:string"
use="optional" />
        <xs:attribute name="external" type="xs:string" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

</xs:complexType>

<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/sip_error_messages_set

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_sip_error_messages_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="sipErrorMessagesOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="sipErrorMessagesOptionsType" external="666:language:en,text:New
desc;language:ru,text:Новое описание|667:language:en,text:Desc"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_sip_error_messages_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="external"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Hc_sip_general - Управление общими свойствами кластера SIP

- [Hc_sip_general_info](#)
- [Hc_sip_general_set](#)
- [Hc_sip_general_clean](#)

[Hc_sip_general_info](#)

Команда позволяет просмотреть общие свойства кластера SIP.

Аналог команды в CoCon:

cluster/adapter/<PA_SIP>/sip/properties/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_general_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show general properties of SIP cluster.
      Usage: http://server:9999/commands/sip\_general\_info.
      Implementation: hc_sip_general_info.erl.
      Git: git@git.eltex.loc:8080/httpTerminal.git.
      Copyright (c) 2018, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- sipGeneralOptionsType -->
  <xs:complexType name="sipGeneralOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="alternative_server" type="stringType" use="optional"/>
        <xs:attribute name="dns_resolve_check_interval" type="integerType" use="optional"
/>
        <xs:attribute name="dns_resolve_default_ttl" type="integerType" use="optional"/>
        <xs:attribute name="expires_alternative_server" type="integerType" use="optional"
/>
        <xs:attribute name="release_ind_ack" type="booleanType" use="optional"/>
        <xs:attribute name="release_ind_ack_timeout" type="integerType" use="optional"/>
        <xs:attribute name="setup_ind_ack_timeout" type="integerType" use="optional"/>
        <xs:attribute name="stat_info_to_log_interval" type="integerType" use="optional"/
>
        <xs:attribute name="tcp_connection_idle_timeout" type="integerType"
use="optional"/>
        <xs:attribute name="tcp_connect_timeout" type="integerType" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="cluster_options" type="clusterInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>

```



```

        <xs:element name="cluster_options" type="clusterInfoInType"
minOccurs="1"/>
        </xs:all>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_general_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="sip_general_info.xsd">
    <request>
        <cluster_options cluster="sip1" />
    </request>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_general_info.xsd">
  <result>
    <cluster_options>
      <cluster name="sip1">
        <options xs:type="sipGeneralOptionsType" dns_resolve_check_interval="600"
          dns_resolve_default_ttl="3600" release_ind_ack="false" release_ind_ack_timeout="2000"
          setup_ind_ack_timeout="5000" stat_info_to_log_interval="0"/>
      </cluster>
    </cluster_options>
  </result>
</out>
```

Нс_sip_general_set

Команда позволяет редактировать общие свойства кластера SIP.

Аналог команды в CoCon:

cluster/adapter/<PA_SIP>/sip/properties/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_general_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show general properties of SIP cluster.
      Usage: http://server:9999/commands/sip_general_info.
      Implementation: hc_sip_general_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2018, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- sipGeneralOptionsType-->
  <xs:complexType name="sipGeneralOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="alternative_server" type="stringType" use="optional"/>
        <xs:attribute name="dns_resolve_check_interval" type="integerType" use="optional"
/>
        <xs:attribute name="dns_resolve_default_ttl" type="integerType" use="optional"/>
        <xs:attribute name="expires_alternative_server" type="integerType" use="optional"
/>
        <xs:attribute name="release_ind_ack" type="booleanType" use="optional"/>
        <xs:attribute name="release_ind_ack_timeout" type="integerType" use="optional"/>
        <xs:attribute name="setup_ind_ack_timeout" type="integerType" use="optional"/>
        <xs:attribute name="stat_info_to_log_interval" type="integerType" use="optional"/
>
        <xs:attribute name="tcp_connection_idle_timeout" type="integerType"
use="optional"/>
        <xs:attribute name="tcp_connect_timeout" type="integerType" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="cluster_options" type="clusterInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>

```

```

        <xs:element name="cluster_options" type="clusterInfoInType"
minOccurs="1"/>
        </xs:all>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_general_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_general_set.xsd">
  <request>
    <cluster_options>
      <cluster name="sip1">
        <options xs:type="sipGeneralOptionsType" dns_resolve_check_interval="500"
dns_resolve_default_ttl="1800" release_ind_ack="true" release_ind_ack_timeout="1000"
setup_ind_ack_timeout="3000" stat_info_to_log_interval="1" />
      </cluster>
    </cluster_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_timers_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="sipTimersOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="sipTimersOptionsType" subscribe_over="5" subscribe_limit="0"
        sip_TC="181000" sip_T4="5000" sip_T2="4000" sip_T1="500" rfc4028_use_se="1800"
        rfc4028_min_se="300" reg_expire_min="90" reg_expire_max="3600" invite_transaction_timeout="32000"
        forbidden="100" find_me="10000"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_sip_general_clean

Команда позволяет вернуть значение по умолчанию для общих свойств кластера SIP.

Аналог команды в CoCon:

cluster/**a**dapter/**<PA_SIP>**/sip/properties/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_general_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean general properties of SIP cluster.
      Usage: http://server:9999/commands/sip_general_clean.
      Implementation: hc_sip_general_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="cluster_options" type="clusterCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="cluster_options" type="clusterCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.23.34:9999/commands/sip_general_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sip_general_clean.xsd">
  <request>
    <cluster_options>
      <cluster name="sip1">
        <option name="dns_resolve_check_interval" />
        <option name="dns_resolve_default_ttl" />
        <option name="setup_init_ttl" />
      </cluster>
    </cluster_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sip_timers_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="find_me"/>
        <success option="forbidden"/>
        <success option="rfc4028_min_se"/>
        <success option="sip_T1"/>
        <success option="sip_T2"/>
        <success option="sip_T4"/>
        <success option="subscribe_over"/>
        <error option="session_limit" reason="{error,not_found}"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_sip_internal_messages - Управление текстовыми описаниями для внутренних кодов SIP

- [sip_internal_messages_clean](#)
- [sip_internal_messages_info](#)
- [sip_internal_messages_set](#)

`sip_internal_messages_clean`

Команда возвращает дефолтные значения текстовых описаний для внутренних кодов SIP.

Аналог команды в CoCon:

`domain/<DOMAIN>/sip/sip-internal-messages/clean`

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/sip-internal-messages_clean

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="sipStatusMessagesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="code_100" type="xs:string" use="optional" />
        <xs:attribute name="code_180" type="xs:string" use="optional" />
        <xs:attribute name="code_181" type="xs:string" use="optional" />
        <xs:attribute name="code_182" type="xs:string" use="optional" />
        <xs:attribute name="code_183" type="xs:string" use="optional" />
        <xs:attribute name="code_200" type="xs:string" use="optional" />
        <xs:attribute name="code_202" type="xs:string" use="optional" />
        <xs:attribute name="code_300" type="xs:string" use="optional" />
        <xs:attribute name="code_301" type="xs:string" use="optional" />
        <xs:attribute name="code_302" type="xs:string" use="optional" />
        <xs:attribute name="code_305" type="xs:string" use="optional" />
        <xs:attribute name="code_380" type="xs:string" use="optional" />
        <xs:attribute name="code_400" type="xs:string" use="optional" />
        <xs:attribute name="code_401" type="xs:string" use="optional" />
        <xs:attribute name="code_402" type="xs:string" use="optional" />
        <xs:attribute name="code_403" type="xs:string" use="optional" />
        <xs:attribute name="code_404" type="xs:string" use="optional" />
        <xs:attribute name="code_405" type="xs:string" use="optional" />
        <xs:attribute name="code_406" type="xs:string" use="optional" />
        <xs:attribute name="code_407" type="xs:string" use="optional" />
        <xs:attribute name="code_408" type="xs:string" use="optional" />
        <xs:attribute name="code_410" type="xs:string" use="optional" />
        <xs:attribute name="code_413" type="xs:string" use="optional" />
        <xs:attribute name="code_414" type="xs:string" use="optional" />
        <xs:attribute name="code_415" type="xs:string" use="optional" />
        <xs:attribute name="code_416" type="xs:string" use="optional" />
        <xs:attribute name="code_420" type="xs:string" use="optional" />
        <xs:attribute name="code_421" type="xs:string" use="optional" />
        <xs:attribute name="code_422" type="xs:string" use="optional" />
        <xs:attribute name="code_423" type="xs:string" use="optional" />
        <xs:attribute name="code_433" type="xs:string" use="optional" />
        <xs:attribute name="code_480" type="xs:string" use="optional" />
        <xs:attribute name="code_481" type="xs:string" use="optional" />
        <xs:attribute name="code_482" type="xs:string" use="optional" />
        <xs:attribute name="code_483" type="xs:string" use="optional" />
        <xs:attribute name="code_484" type="xs:string" use="optional" />
        <xs:attribute name="code_485" type="xs:string" use="optional" />
        <xs:attribute name="code_486" type="xs:string" use="optional" />
        <xs:attribute name="code_487" type="xs:string" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

        <xs:attribute name="code_488" type="xs:string" use="optional" />
        <xs:attribute name="code_489" type="xs:string" use="optional" />
        <xs:attribute name="code_491" type="xs:string" use="optional" />
        <xs:attribute name="code_493" type="xs:string" use="optional" />
        <xs:attribute name="code_500" type="xs:string" use="optional" />
        <xs:attribute name="code_501" type="xs:string" use="optional" />
        <xs:attribute name="code_502" type="xs:string" use="optional" />
        <xs:attribute name="code_503" type="xs:string" use="optional" />
        <xs:attribute name="code_504" type="xs:string" use="optional" />
        <xs:attribute name="code_505" type="xs:string" use="optional" />
        <xs:attribute name="code_513" type="xs:string" use="optional" />
        <xs:attribute name="code_600" type="xs:string" use="optional" />
        <xs:attribute name="code_603" type="xs:string" use="optional" />
        <xs:attribute name="code_604" type="xs:string" use="optional" />
        <xs:attribute name="code_606" type="xs:string" use="optional" />
        <xs:attribute name="external" type="xs:string" use="optional" />
    </xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- resultType-->
<xs:complexType name="resultType">
    <xs:all>
        <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
</xs:complexType>

<!-- in-->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainSetInType"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>

<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
            <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_internal_messages_clean

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_sip_internal_messages_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="unsupportedMedia"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sip_internal_messages_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="unsupportedMedia"/>
      </domain>
    </domain_options>
  </result>
</out>
```

[sip_internal_messages_info](#)

Команда возвращает текстовые описания для SIP-cause

Аналог команды в CoCon:**domain/<DOMAIN>/sip/sip-error-messages/info****Метод http запроса:**

POST

Шаблон http запроса:http://host:port/commands/sip_internal-messages_info

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="sipInternalMessagesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="normal" type="xs:string" use="optional"
/>
        <xs:attribute name="originationDenied" type="xs:string" use="optional"
/>
        <xs:attribute name="authorisationFailure" type="xs:string" use="optional"
/>
        <xs:attribute name="unsupportedMedia" type="xs:string" use="optional"
/>
        <xs:attribute name="aPtyAbandon" type="xs:string" use="optional"
/>
        <xs:attribute name="invalidCollectedInformation" type="xs:string" use="optional"
/>
        <xs:attribute name="collectInformationFailure" type="xs:string" use="optional"
/>
        <xs:attribute name="aPtyDisc" type="xs:string" use="optional"
/>
        <xs:attribute name="bPtyDisc" type="xs:string" use="optional"
/>
        <xs:attribute name="routeSelectFailure" type="xs:string" use="optional"
/>
        <xs:attribute name="routeFailure1" type="xs:string" use="optional"
/>
        <xs:attribute name="routeFailure2" type="xs:string" use="optional"
/>
        <xs:attribute name="invalidNumber" type="xs:string" use="optional"
/>
        <xs:attribute name="unassignedNumber" type="xs:string" use="optional"
/>
        <xs:attribute name="numberIncomplete" type="xs:string" use="optional"
/>
        <xs:attribute name="oNoAnswer" type="xs:string" use="optional"
/>
        <xs:attribute name="bPtyNoAnswer" type="xs:string" use="optional"
/>
        <xs:attribute name="noCircuitAvailable" type="xs:string" use="optional"
/>
        <xs:attribute name="terminationDenied" type="xs:string" use="optional"
/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

/>         <xs:attribute name="notReachable"                type="xs:string" use="optional"
/>         <xs:attribute name="bPtyBusyUDUB"              type="xs:string" use="optional"
/>         <xs:attribute name="bPtyBusyNDUB"              type="xs:string" use="optional"
/>         <xs:attribute name="ss7Failure"                 type="xs:string" use="optional"
/>         <xs:attribute name="calledPartyRejected"       type="xs:string" use="optional"
/>         <xs:attribute name="tException"                 type="xs:string" use="optional"
/>         <xs:attribute name="systemFailure"              type="xs:string" use="optional"
/>         <xs:attribute name="tooManyHops"                type="xs:string" use="optional"
/>         <xs:attribute name="heartbeat"                  type="xs:string" use="optional"
/>         <xs:attribute name="heartbeat_timeout"          type="xs:string" use="optional"
/>         <xs:attribute name="heartbeat_timeout_check"    type="xs:string" use="optional"
/>         <xs:attribute name="session_timeout"            type="xs:string" use="optional"
/>         <xs:attribute name="conversationTimeout"        type="xs:string" use="optional"
/>         <xs:attribute name="uaPreemption"               type="xs:string" use="optional"
/>         <xs:attribute name="reservedResourcesPreemted" type="xs:string" use="optional"
/>         <xs:attribute name="genericPreemption"          type="xs:string" use="optional"
/>         <xs:attribute name="nonIpPreemption"            type="xs:string" use="optional"
/>         <xs:attribute name="fork"                       type="xs:string" use="optional"
/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>

```

```

        <xs:element name="request">
            <xs:complexType>
                <xs:all>
                    <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1"/>
                </xs:all>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_internal_messages_info

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_sip_internal_messages_info.xsd">
  <request>
    <domain_options domain="biysk.local" />
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sip_internal_messages_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="sipInternalMessagesOptionsType"/>
      </system>
      <domain name="biysk.local">

```

```

<options xs:type="sipInternalMessagesOptionsType"
unsupportedMedia="language:default,text:Unsupported media;language:en,text:Unsupported
media;language:ru,text:Неподдерживаемый тип меди" unassignedNumber="language:default,text:Called
number is wrong;language:en,text:Called number is wrong;language:ru,text:Вызываемый номер -
неверный" uaPreemption="language:default,text:UA Preemption;language:en,text:UA
Preemption;language:ru,text:UA Preemption" tooManyHops="language:default,text:Too Many
Hops;language:en,text:Too Many Hops;language:ru,text:Слишком много переходов"
terminationDenied="language:default,text:Call process is terminated;language:en,text:Call process
is terminated;language:ru,text:Процесс вызова прерван" tException="language:default,text:Internal
error;language:en,text:Internal error;language:ru,text:Внутренняя ошибка"
systemFailure="language:default,text:Internal error;language:en,text:Internal
error;language:ru,text:Внутренняя ошибка" ss7Failure="language:default,text:SS7
failure;language:en,text:SS7 failure;language:ru,text:"
session_timeout="language:default,text:Conversation timeout;language:en,text:Conversation
timeout;language:ru,text:Истекло время разговора"
routeSelectFailure="language:default,text:Called number is wrong;language:en,text:Called number
is wrong;language:ru,text:Вызываемый номер - неверный"
routeFailure2="language:default,text:Called number is wrong;language:en,text:Called number is
wrong;language:ru,text:Вызываемый номер - неверный" routeFailure1="language:default,text:Called
number is wrong;language:en,text:Called number is wrong;language:ru,text:Вызываемый номер -
неверный" reservedResourcesPreemted="language:default,text:Reserved Resources
Preempted;language:en,text:Reserved Resources Preempted;language:ru,text:Reserved Resources
Preempted" originationDenied="language:default,text:Call is broken in origination's
leg;language:en,text:Call is broken in origination's leg;language:ru,text:"
oNoAnswer="language:default,text:No answer;language:en,text:No answer;language:ru,text:Нет
ответа" numberIncomplete="language:default,text:Address Incomplete;language:en,text:Address
Incomplete;language:ru,text:Неполный адрес" notReachable="language:default,text:Destination is
unreachable;language:en,text:Destination is unreachable;language:ru,text:Конечный пункт
недостижим" normal="language:default,text:Call is complete;language:en,text:Call is
complete;language:ru,text:Вызов завершен" nonIpPreemption="language:default,text:Non-IP
Preemption;language:en,text:Non-IP Preemption;language:ru,text:Non-IP Preemption"
noCircuitAvailable="language:default,text:Temporarily unavailable;language:en,text:Temporarily
unavailable;language:ru,text:Временно недоступен" invalidNumber="language:default,text:Called
number is wrong;language:en,text:Called number is wrong;language:ru,text:Вызываемый номер -
неверный" invalidCollectedInformation="language:default,text:Invalid collected
information;language:en,text:Invalid collected information;language:ru,text:"
heartbeat_timeout_check="language:default,text:Remote session is
inaccessible;language:en,text:Remote session is inaccessible;language:ru,text:Удаленный сеанс
недоступен" heartbeat_timeout="language:default,text:Remote session is
inaccessible;language:en,text:Remote session is inaccessible;language:ru,text:Удаленный сеанс
недоступен" heartbeat="language:default,text:Remote session is
inaccessible;language:en,text:Remote session is inaccessible;language:ru,text:Удаленный сеанс
недоступен" genericPreemption="language:default,text:Generic Preemption;language:en,text:Generic
Preemption;language:ru,text:Generic Preemption" fork="language:default,text:Forked leg is
cancelled;language:en,text:Forked leg is cancelled;language:ru,text:Forked leg is cancelled"
conversationTimeout="language:default,text:Conversation timeout;language:en,text:Conversation
timeout;language:ru,text:Истекло время разговора"
collectInformationFailure="language:default,text:Collect information
failure;language:en,text:Collect information failure;language:ru,text:"
calledPartyRejected="language:default,text:Call is rejected by destination's
leg;language:en,text:Call is rejected by destination's leg;language:ru,text:"

```



```

bPtyNoAnswer="language:default,text:No answer;language:en,text:No answer;language:ru,text:Нет
ответа" bPtyDisc="language:default,text:Disconnect from B side;language:en,text:Disconnect from B
side;language:ru,text:Отсоединение со стороны B" bPtyBusyUDUB="language:default,text:User is
busy;language:en,text:User is busy;language:ru,text:Абонент занят"
bPtyBusyNDUB="language:default,text:Destination is Busy;language:en,text:Destination is
Busy;language:ru,text:Конечный пункт занят"
authorisationFailure="language:default,text:Authorisation failure;language:en,text:Authorisation
failure;language:ru,text:Авторизация неуспешна" aPtyDisc="language:default,text:Disconnect from A
side;language:en,text:Disconnect from A side;language:ru,text:Отсоединение со стороны A"
aPtyAbandon="language:default,text:Call is cancelled by A side;language:en,text:Call is
cancelled by A side;language:ru,text:Вызов отменен стороной A"/>
    </domain>
  </domain_options>
</result>
</out>

```

[sip_internal_messages_set](#)

Команда задает значения текстовых описаний для SIP-cause

Аналог команды в CoCon:

domain/<DOMAIN>/sip/sip-error-messages/set

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/sip_internal_messages_set

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="sipErrorMessagesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="code_0" type="xs:string" use="optional" />
        <xs:attribute name="code_1" type="xs:string" use="optional" />
        <xs:attribute name="code_2" type="xs:string" use="optional" />
        <xs:attribute name="code_3" type="xs:string" use="optional" />
        <xs:attribute name="code_4" type="xs:string" use="optional" />
        <xs:attribute name="code_5" type="xs:string" use="optional" />
        <xs:attribute name="code_6" type="xs:string" use="optional" />
        <xs:attribute name="code_7" type="xs:string" use="optional" />
        <xs:attribute name="code_8" type="xs:string" use="optional" />
        <xs:attribute name="code_9" type="xs:string" use="optional" />
        <xs:attribute name="code_10" type="xs:string" use="optional" />
        <xs:attribute name="code_11" type="xs:string" use="optional" />
        <xs:attribute name="code_12" type="xs:string" use="optional" />
        <xs:attribute name="code_14" type="xs:string" use="optional" />
        <xs:attribute name="code_15" type="xs:string" use="optional" />
        <xs:attribute name="code_16" type="xs:string" use="optional" />
        <xs:attribute name="network_not_configured" type="xs:string" use="optional" />
        <xs:attribute name="alias_declaration_error" type="xs:string" use="optional" />
        <xs:attribute name="invalid_alias_digits" type="xs:string" use="optional" />
        <xs:attribute name="wrong_identififier" type="xs:string" use="optional" />
        <xs:attribute name="bad_template" type="xs:string" use="optional" />
        <xs:attribute name="name_is_busy" type="xs:string" use="optional" />
        <xs:attribute name="unknown_password" type="xs:string" use="optional" />
        <xs:attribute name="unknown_login" type="xs:string" use="optional" />
        <xs:attribute name="no_ldap_connection" type="xs:string" use="optional" />
        <xs:attribute name="execution_error" type="xs:string" use="optional" />
        <xs:attribute name="licence_limit" type="xs:string" use="optional" />
        <xs:attribute name="bad_value" type="xs:string" use="optional" />
        <xs:attribute name="wrong_set" type="xs:string" use="optional" />
        <xs:attribute name="cancelled" type="xs:string" use="optional" />
        <xs:attribute name="error_length" type="xs:string"
use="optional" />
        <xs:attribute name="error_contact_already_declared" type="xs:string"
use="optional" />
        <xs:attribute name="error_channel_id_not_found" type="xs:string"
use="optional" />
        <xs:attribute name="external" type="xs:string" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

</xs:complexType>

<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_internal_messages_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_sip_internal_messages_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="sipInternalMessagesOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="sipInternalMessagesOptionsType"
unsupportedMedia="language:ru,text:Неподдерживаемый тип медиа"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_sip_internal_messages_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="unsupportedMedia"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_sip_ldap - Управление параметрами соединения с сервером LDAP

- [Нс_sip_ldap_info](#)
- [Нс_sip_ldap_set](#)
- [Нс_sip_ldap_clean](#)

Нс_sip_ldap_info

Команда позволяет просмотреть свойства LDAP кластера SIP.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/ldap/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_ldap_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show LDAP properties of SIP cluster.
      Usage: http://server:9999/commands/sip_ldap_info.
      Implementation: hc_sip_ldap_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- sipLDAPOptionsType -->
  <xs:complexType name="sipLDAPOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="ip" type="stringType" use="optional"/>
        <xs:attribute name="port" type="stringType" use="optional"/>
        <xs:attribute name="access.path" type="stringType" use="optional"/>
        <xs:attribute name="access.login" type="stringType" use="optional"/>
        <xs:attribute name="access.passw" type="stringType" use="optional"/>
        <xs:attribute name="attr.domain" type="stringType" use="optional"/>
        <xs:attribute name="attr.phone" type="stringType" use="optional"/>
        <xs:attribute name="attr.login" type="stringType" use="optional"/>
        <xs:attribute name="attr.passw" type="stringType" use="optional"/>
        <xs:attribute name="alive_timeout"
          type="xs:nonNegativeInteger"
          use="optional"/>
        <xs:attribute name="enable" type="xs:boolean" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1"/>

```

```
        </xs:all>
        </xs:complexType>
    </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>
```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/sip_ldap_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="sip_ldap_info.xsd">
    <request>
        <domain_options domain="biysk.local" />
    </request>
</in>
```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_ldap_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="sipLDAPOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="sipLDAPOptionsType" port="389" ip="localhost" enable="true"
        attr.phone="telephoneNumber" attr.passw="description" attr.login="cn" attr.domain="dc"
        alive_timeout="10000" access.path="ou=sip1,dc=ecss" access.passw="admin"
        access.login="cn=admin,dc=ecss"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_sip_ldap_set

Команда позволяет редактировать свойства LDAP кластера SIP.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/ldap/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_ldap_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set LDAP properties of SIP cluster.
      Usage: http://server:9999/commands/sip_ldap_set.
      Implementation: hc_sip_ldap_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- msExtensionType -->
  <xs:simpleType name="msExtensionType">
    <xs:restriction base="unionType">
      <xs:enumeration value="header"/>
      <xs:enumeration value="body"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- limitType -->
  <xs:simpleType name="limitType">
    <xs:restriction base="unionType">
      <xs:enumeration value="none"/>
      <xs:enumeration value="normal"/>
      <xs:enumeration value="warning"/>
      <xs:enumeration value="minor"/>
      <xs:enumeration value="major"/>
      <xs:enumeration value="critical"/>
      <xs:enumeration value="fatal"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- sipLDAPOptionsType-->
  <xs:complexType name="sipLDAPOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="ip" type="stringType" use="optional"/>
        <xs:attribute name="port" type="stringType" use="optional"/>
        <xs:attribute name="access.path" type="stringType" use="optional"/>
        <xs:attribute name="access.login" type="stringType" use="optional"/>
        <xs:attribute name="access.passw" type="stringType" use="optional"/>
        <xs:attribute name="attr.domain" type="stringType" use="optional"/>
        <xs:attribute name="attr.phone" type="stringType" use="optional"/>
        <xs:attribute name="attr.login" type="stringType" use="optional"/>
        <xs:attribute name="attr.passw" type="stringType" use="optional"/>
        <xs:attribute name="alive_timeout"
          type="xs:nonNegativeInteger"
          use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

        <xs:attribute name="enable" type="xs:boolean" use="optional"/>
    </xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- resultType-->
<xs:complexType name="resultType">
    <xs:all>
        <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
</xs:complexType>
<!-- in-->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainSetInType"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
            <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_ldap_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_ldap_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="sipLDAPOptionsType"/>
      </system>
      <domain name="ct.office">
        <options xs:type="sipLDAPOptionsType" ip="192.168.1.21" port="3890"
        access.path="ou=pa_sip,dc=ecss" access.login="cn=ecss_system,dc=ecss" access.passw="admin"
        attr.domain="dc" attr.phone="phoneNum" attr.login="cn" attr.passw="password1" enable="true" />
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_ldap_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="access.login"/>
        <success option="access.passw"/>
        <success option="access.path"/>
        <success option="attr.domain"/>
        <success option="attr.login"/>
        <success option="attr.passw"/>
        <success option="attr.phone"/>
        <success option="enable"/>
        <success option="ip"/>
        <success option="port"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_sip_ldap_clean

Команда позволяет установить значение по умолчанию свойства LDAP кластера SIP.

Аналог команды в CoCon:

`domain/<DOMAIN>/sip/ldap/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/sip_ldap_clean`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean LDAP properties of SIP cluster.
      Usage: http://server:9999/commands/sip_ldap_clean.
      Implementation: hc_sip_ldap_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_ldap_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sip_ldap_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="ct.office">
        <option name="ip"/>
        <option name="port"/>
        <option name="access.path"/>
        <option name="access.login"/>
        <option name="access.passw"/>
        <option name="attr.domain"/>
        <option name="attr.phone"/>
        <option name="attr.login"/>
        <option name="attr.passw"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_ldap_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="ct.office">
        <success option="access.login"/>
        <success option="access.passw"/>
        <success option="access.path"/>
        <success option="attr.domain"/>
        <success option="attr.login"/>
        <success option="attr.passw"/>
        <success option="attr.phone"/>
        <success option="ip"/>
        <success option="port"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Hc_sip_number_modifications - Формирование правил модификации номера

- [Hc_sip_number_modifications_clean](#)
- [Hc_sip_number_modifications_info](#)
- [Hc_sip_number_modifications_set](#)

[Hc_sip_number_modifications_clean](#)

Команда сброса настроек SIP-модификаторов.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/modifications/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_number_modifications_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean properties of sip number modifications.
      Usage: http://server:9999/commands/sip_number_modifications_clean.
      Implementation: hc_sip_number_modifications_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_number_modifications_clean


```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sip_number_modifications_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="list"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sip_number_modifications_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="list"/>
      </domain>
    </domain_options>
  </result>
</out>

```

[Hc_sip_number_modifications_info](#)

Команда просмотра настроек SIP-модификаторов.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/modifications/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_number_modifications_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of sip number modifications.
      Usage: http://server:9999/commands/sip_number_modifications_info.
      Implementation: hc_sip_number_modifications_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="sipNumberModificationsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="list" type="xs:string" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainInfoInType" minOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_number_modifications_info

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_sip_number_modifications_info.xsd">
<request>
  <domain_options domain="biysk.local"/>
</request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sip_number_modifications_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="sipNumberModificationsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="sipNumberModificationsType" list=""rule1","cgpn","222*","$","--
-$","rule2","cgpn","xx8x.","$","+(777)$""/>
      </domain>
    </domain_options>
  </result>
</out>

```

hc_sip_number_modifications_set

Команда настройки SIP-модификаторов.

Аналог команды в CoCon:**domain/<DOMAIN>/sip/modifications/set****Метод HTTP-запроса:**

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_modifications_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set sip number modifications
      Usage: http://server:9999/commands/sip_number_modifications_set.
      Implementation: hc_sip_number_modifications_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="sipNumberModificationsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="list" type="xs:string" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

```
http://192.168.1.21:9999/commands/sip_modifications_set
```

```

<?xml version="1.0"?>
<in
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_sip_number_modifications_set.xsd">
<request>
<domain_options>
<system>
<options xs:type="sipNumberModificationsType"/>
</system>
<domain name="biysk.local">
<options xs:type="sipNumberModificationsType"
list=""rule1","cgpn","222*","$","---$";rule2","cgpn","xx8x.","$","+ (777)$""/
>
</domain>
</domain_options>
</request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<in
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_sip_number_modifications_set.xsd">
<request>
<domain_options>
<system>
<options xs:type="sipNumberModificationsType"/>
</system>
<domain name="biysk.local">
<options xs:type="sipNumberModificationsType"
list=""rule1","cgpn","222*","$","---$";rule2","cgpn","xx8x.","$","+ (777)$""/
>
</domain>
</domain_options>
</request>
</in>

```

Hc_sip_pcap_trace - Управление трассировкой

- [Hc_sip_pcap_trace_start](#)
- [Hc_sip_pcap_trace_stop](#)

Hc_sip_pcap_trace_start

Команда позволяет начать запись tcp-dump на SIP-адаптере.

Аналог команды в CoSop:

cluster/adapter/<PA_SIP>/pcap-trace/start

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_pcap_trace_start

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки;
- 404 – в случае непредвиденной ошибки.

XSD-схема XML-файла с запросом/ответом:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- in -->
  <xs:element name = "in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="id" use="required" type="xs:string">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  Unique identifier of the pcap trace. Used to stop current
pcap trace.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="cluster" use="required" type="xs:string">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  SIP adapter's cluster name.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="node" use="required" type="xs:string"/>
            <xs:attribute name="ip" use="optional" type="xs:string">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  One of the properties ip, ip_set must be defiled.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="ip_set" use="optional" type="xs:string">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  One of the properties ip, ip_set must be defiled.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="port" use="optional" type="xs:string"/>
            <xs:attribute name="reply_host" use="required" type="xs:string"/>
            <xs:attribute name="reply_port" use="required" type="xs:integer"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->

```

```

<xs:element name = "out">
  <xs:complexType>
    <xs:all>
      <xs:element name="error" type="errorType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1:

Запрос: http://192.168.1.21:9999/commands/sip_pcap_trace_start

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request port="" reply_port="51679" ip="192.168.2.61" node="sip1@ecss1" cluster="sip1"
  reply_host="192.168.2.21" id="51679"/>
</in>

```

Ответ: 201

Пример 2:

Запрос:

http://192.168.1.21:9999/commands/sip_pcap_trace_start

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request port="" reply_port="4444" ip="192.168.2.21" node="sip1@ecss1" cluster="sip1"
  reply_host="192.168.2.21" id="4444"/>
</in>

```

Ответ: 201

[Hc_sip_pcap_trace_stop](#)

Команда позволяет остановить запись tcp-dump на SIP-адаптере.

Аналог команды в CoSop:

cluster/adapter/<PA_SIP>/pcap-trace/stop

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_pcap_trace_stop

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае непредвиденной ошибки.

XSD-схема XML-файла с запросом/ответом:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:element name = "in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="id" use="required" type="xs:string">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  Unique identifier of the pcap trace. Used to stop current
pcap trace.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="cluster" use="required" type="xs:string">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  SIP adapter's cluster name.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_pcap_trace_stop

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_pcap_trace_stop.xsd">
  <request id="4444" cluster="sip1" />
</in>
```

Ответ: 204

Hc_sip_status_messages - Управление текстовыми описаниями для SIP-cause

- [Hc_sip_status_messages_clean](#)
- [Hc_sip_status_messages_info](#)
- [Hc_sip_status_messages_set](#)

[Hc_sip_status_messages_clean](#)

Команда возвращает дефолтные значения текстовых описаний для SIP статусов

Аналог команды в CoCon:

domain/<DOMAIN>/sip/sip-status-messages/clean

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/sip-status-messages_clean

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_status_messages_clean

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_sip_status_messages_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="code_100"/>
        <option name="external"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sip_status_messages_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="code_100"/>
        <success option="external"/>
      </domain>
    </domain_options>
  </result>
</out>
```

[Hc_sip_status_messages_info](#)

Команда возвращает текстовые описания для SIP-cause.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/sip-status-messages/info

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/sip-status-messages_info

Код http ответа

200 – в случае успеха;
404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="sipStatusMessagesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="code_100" type="xs:string" use="optional" />
        <xs:attribute name="code_180" type="xs:string" use="optional" />
        <xs:attribute name="code_181" type="xs:string" use="optional" />
        <xs:attribute name="code_182" type="xs:string" use="optional" />
        <xs:attribute name="code_183" type="xs:string" use="optional" />
        <xs:attribute name="code_200" type="xs:string" use="optional" />
        <xs:attribute name="code_202" type="xs:string" use="optional" />
        <xs:attribute name="code_300" type="xs:string" use="optional" />
        <xs:attribute name="code_301" type="xs:string" use="optional" />
        <xs:attribute name="code_302" type="xs:string" use="optional" />
        <xs:attribute name="code_305" type="xs:string" use="optional" />
        <xs:attribute name="code_380" type="xs:string" use="optional" />
        <xs:attribute name="code_400" type="xs:string" use="optional" />
        <xs:attribute name="code_401" type="xs:string" use="optional" />
        <xs:attribute name="code_402" type="xs:string" use="optional" />
        <xs:attribute name="code_403" type="xs:string" use="optional" />
        <xs:attribute name="code_404" type="xs:string" use="optional" />
        <xs:attribute name="code_405" type="xs:string" use="optional" />
        <xs:attribute name="code_406" type="xs:string" use="optional" />
        <xs:attribute name="code_407" type="xs:string" use="optional" />
        <xs:attribute name="code_408" type="xs:string" use="optional" />
        <xs:attribute name="code_410" type="xs:string" use="optional" />
        <xs:attribute name="code_413" type="xs:string" use="optional" />
        <xs:attribute name="code_414" type="xs:string" use="optional" />
        <xs:attribute name="code_415" type="xs:string" use="optional" />
        <xs:attribute name="code_416" type="xs:string" use="optional" />
        <xs:attribute name="code_420" type="xs:string" use="optional" />
        <xs:attribute name="code_421" type="xs:string" use="optional" />
        <xs:attribute name="code_422" type="xs:string" use="optional" />
        <xs:attribute name="code_423" type="xs:string" use="optional" />
        <xs:attribute name="code_433" type="xs:string" use="optional" />
        <xs:attribute name="code_480" type="xs:string" use="optional" />
        <xs:attribute name="code_481" type="xs:string" use="optional" />
        <xs:attribute name="code_482" type="xs:string" use="optional" />
        <xs:attribute name="code_483" type="xs:string" use="optional" />
        <xs:attribute name="code_484" type="xs:string" use="optional" />
        <xs:attribute name="code_485" type="xs:string" use="optional" />
        <xs:attribute name="code_486" type="xs:string" use="optional" />
        <xs:attribute name="code_487" type="xs:string" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```



```

        <xs:attribute name="code_488" type="xs:string" use="optional" />
        <xs:attribute name="code_489" type="xs:string" use="optional" />
        <xs:attribute name="code_491" type="xs:string" use="optional" />
        <xs:attribute name="code_493" type="xs:string" use="optional" />
        <xs:attribute name="code_500" type="xs:string" use="optional" />
        <xs:attribute name="code_501" type="xs:string" use="optional" />
        <xs:attribute name="code_502" type="xs:string" use="optional" />
        <xs:attribute name="code_503" type="xs:string" use="optional" />
        <xs:attribute name="code_504" type="xs:string" use="optional" />
        <xs:attribute name="code_505" type="xs:string" use="optional" />
        <xs:attribute name="code_513" type="xs:string" use="optional" />
        <xs:attribute name="code_600" type="xs:string" use="optional" />
        <xs:attribute name="code_603" type="xs:string" use="optional" />
        <xs:attribute name="code_604" type="xs:string" use="optional" />
        <xs:attribute name="code_606" type="xs:string" use="optional" />
        <xs:attribute name="external" type="xs:string" use="optional" />
    </xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- resultType-->
<xs:complexType name="resultType">
    <xs:all>
        <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
</xs:complexType>

<!-- in-->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>

<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>

```

```
</xs:schema>
```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_status_messages_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_sip_status_messages_info.xsd">
  <request>
    <domain_options domain="biysk.local" />
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sip_status_messages_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="sipStatusMessagesOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="sipStatusMessagesOptionsType" external=""
code_606="language:default,text:Not Acceptable;language:en,text:Not Acceptable;language:ru,text:"
code_604="language:default,text:Does not exist anywhere;language:en,text:Does not exist
anywhere;language:ru,text:Абонент не существует"
code_603="language:default,text:Decline;language:en,text:Decline;language:ru,text:Вызов отклонён"
code_600="language:default,text:Busy Everywhere;language:en,text:Busy
Everywhere;language:ru,text:Абонент не желает принимать вызов"
code_513="language:default,text:Message Too Large;language:en,text:Message Too
Large;language:ru,text:Размер сообщения слишком велик" code_505="language:default,text:SIP
Version not supported;language:en,text:SIP Version not supported;language:ru,text:Версия SIP не
поддерживается" code_504="language:default,text:Server Time-out;language:en,text:Server Time-
out;language:ru,text:Истекло время ожидания ответа" code_503="language:default,text:Service
Unavailable;language:en,text:Service Unavailable;language:ru,text:Обслуживание недоступно"
code_502="language:default,text:Bad Gateway;language:en,text:Bad Gateway;language:ru,text:"
code_501="language:default,text:Not Implemented;language:en,text:Not
Implemented;language:ru,text:Не реализовано" code_500="language:default,text:Internal Server
Error;language:en,text:Internal Server Error;language:ru,text:Внутренняя ошибка сервера"
code_493="language:default,text:Undecipherable;language:en,text:Undecipherable;language:ru,text:
Невозможно дешифровать" code_491="language:default,text:Request Pending;language:en,text:Request
Pending;language:ru,text:Запрос отложен" code_489="language:default,text:Bad
Event;language:en,text:Bad Event;language:ru,text:Неизвестное событие"
```

```

code_488="language:default,text:Not Acceptable Here;language:en,text:Not Acceptable
Here;language:ru,text:Запрос не принят" code_487="language:default,text:Request
Terminated;language:en,text:Request Terminated;language:ru,text:Запрос отменён"
code_486="language:default,text:Busy Here;language:en,text:Busy Here;language:ru,text:Абонент
занят" code_485="language:default,text:Ambiguous;language:en,text:Ambiguous;language:ru,text:Неод
нозначный адрес" code_484="language:default,text:Address Incomplete;language:en,text:Address
Incomplete;language:ru,text:Неполный адрес" code_483="language:default,text:Too Many
Hops;language:en,text:Too Many Hops;language:ru,text:Превышено допустимое число переходов"
code_482="language:default,text:Loop Detected;language:en,text:Loop
Detected;language:ru,text:Обнаружен замкнутый маршрут" code_481="language:default,text:Call Leg/
Transaction Does Not Exist;language:en,text:Call Leg/Transaction Does Not
Exist;language:ru,text:Действие не выполнено" code_480="language:default,text:Temporarily not
available;language:en,text:Temporarily not available;language:ru,text:Временно недоступен"
code_433="language:default,text:Anonymity Disallowed;language:en,text:Anonymity
Disallowed;language:ru,text:" code_423="language:default,text:Interval Too
Brief;language:en,text:Interval Too Brief;language:ru,text:Слишком короткий интервал"
code_422="language:default,text:Session Interval Too Small;language:en,text:Session Interval Too
Small;language:ru,text:Слишком короткий интервал между сессиями"
code_421="language:default,text:Extension Required;language:en,text:Extension
Required;language:ru,text:Требуется расширение" code_420="language:default,text:Bad
Extension;language:en,text:Bad Extension;language:ru,text:Неизвестное расширение"
code_416="language:default,text:Unsupported URI Scheme;language:en,text:Unsupported URI
Scheme;language:ru,text:Неизвестный тип URI" code_415="language:default,text:Unsupported Media
Type;language:en,text:Unsupported Media Type;language:ru,text:Неподдерживаемый тип меди"
code_414="language:default,text:Request-URI Too Large;language:en,text:Request-URI Too
Large;language:ru,text:Размер URI слишком велик" code_413="language:default,text:Request Entity
Too Large;language:en,text:Request Entity Too Large;language:ru,text:Размер запроса слишком
велик" code_410="language:default,text:Gone;language:en,text:Gone;language:ru,text:"
code_408="language:default,text:Request Timeout;language:en,text:Request
Timeout;language:ru,text:Время обработки запроса истекло" code_407="language:default,text:Proxy
Authentication Required;language:en,text:Proxy Authentication Required;language:ru,text:Требуется
аутентификация на прокис-сервере" code_406="language:default,text:Not
Acceptable;language:en,text:Not Acceptable;language:ru,text:Не доступен"
code_405="language:default,text:Method Not Allowed;language:en,text:Method Not
Allowed;language:ru,text:Метод не поддерживается" code_404="language:default,text:Not
Found;language:en,text:Not Found;language:ru,text:Не найдено"
code_403="language:default,text:Forbidden;language:en,text:Forbidden;language:ru,text:Запрещено"
code_402="language:default,text:Payment Required;language:en,text:Payment
Required;language:ru,text:Требуется оплата"
code_401="language:default,text:Unauthorized;language:en,text:Unauthorized;language:ru,text:Не
авторизован" code_400="language:default,text:Bad Request;language:en,text:Bad
Request;language:ru,text:Некорректный запрос" code_380="language:default,text:Alternative
Service;language:en,text:Alternative Service;language:ru,text:Альтернативные услуги"
code_305="language:default,text:Use Proxy;language:en,text:Use
Proxy;language:ru,text:Использование прокси" code_302="language:default,text:Moved
Temporarily;language:en,text:Moved Temporarily;language:ru,text:Временно не находится по адресу"
code_301="language:default,text:Moved Permanently;language:en,text:Moved
Permanently;language:ru,text:Не находится по адресу" code_300="language:default,text:Multiple
Choices;language:en,text:Multiple Choices;language:ru,text:Выбор из набора"
code_202="language:default,text:Accepted;language:en,text:Accepted;language:ru,text:Запрос
принят" code_200="language:default,text:OK;language:en,text:OK;language:ru,text:OK"

```

```
code_183="language:default,text:Session Progress;language:en,text:Session
Progress;language:ru,text:Прогресс" code_182="language:default,text:Queued;language:en,text:Queue
d;language:ru,text:В очереди" code_181="language:default,text:Call Is Being
Forwarded;language:en,text:Call Is Being Forwarded;language:ru,text:Переадресация вызова"
code_180="language:default,text:Ringing;language:en,text:Ringing;language:ru,text:Вызов"
code_100="language:default,text:Trying;language:en,text:Trying;language:ru,text:Обработка"/>
    </domain>
  </domain_options>
</result>
</out>
```

Ис sip_status_messages_set

Команда задает значения текстовых описаний для SIP-cause.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/sip-status-messages/set

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/sip_status_messages_set

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="sipStatusMessagesOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="code_100" type="xs:string" use="optional" />
        <xs:attribute name="code_180" type="xs:string" use="optional" />
        <xs:attribute name="code_181" type="xs:string" use="optional" />
        <xs:attribute name="code_182" type="xs:string" use="optional" />
        <xs:attribute name="code_183" type="xs:string" use="optional" />
        <xs:attribute name="code_200" type="xs:string" use="optional" />
        <xs:attribute name="code_202" type="xs:string" use="optional" />
        <xs:attribute name="code_300" type="xs:string" use="optional" />
        <xs:attribute name="code_301" type="xs:string" use="optional" />
        <xs:attribute name="code_302" type="xs:string" use="optional" />
        <xs:attribute name="code_305" type="xs:string" use="optional" />
        <xs:attribute name="code_380" type="xs:string" use="optional" />
        <xs:attribute name="code_400" type="xs:string" use="optional" />
        <xs:attribute name="code_401" type="xs:string" use="optional" />
        <xs:attribute name="code_402" type="xs:string" use="optional" />
        <xs:attribute name="code_403" type="xs:string" use="optional" />
        <xs:attribute name="code_404" type="xs:string" use="optional" />
        <xs:attribute name="code_405" type="xs:string" use="optional" />
        <xs:attribute name="code_406" type="xs:string" use="optional" />
        <xs:attribute name="code_407" type="xs:string" use="optional" />
        <xs:attribute name="code_408" type="xs:string" use="optional" />
        <xs:attribute name="code_410" type="xs:string" use="optional" />
        <xs:attribute name="code_413" type="xs:string" use="optional" />
        <xs:attribute name="code_414" type="xs:string" use="optional" />
        <xs:attribute name="code_415" type="xs:string" use="optional" />
        <xs:attribute name="code_416" type="xs:string" use="optional" />
        <xs:attribute name="code_420" type="xs:string" use="optional" />
        <xs:attribute name="code_421" type="xs:string" use="optional" />
        <xs:attribute name="code_422" type="xs:string" use="optional" />
        <xs:attribute name="code_423" type="xs:string" use="optional" />
        <xs:attribute name="code_433" type="xs:string" use="optional" />
        <xs:attribute name="code_480" type="xs:string" use="optional" />
        <xs:attribute name="code_481" type="xs:string" use="optional" />
        <xs:attribute name="code_482" type="xs:string" use="optional" />
        <xs:attribute name="code_483" type="xs:string" use="optional" />
        <xs:attribute name="code_484" type="xs:string" use="optional" />
        <xs:attribute name="code_485" type="xs:string" use="optional" />
        <xs:attribute name="code_486" type="xs:string" use="optional" />
        <xs:attribute name="code_487" type="xs:string" use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

    <xs:attribute name="code_488" type="xs:string" use="optional" />
    <xs:attribute name="code_489" type="xs:string" use="optional" />
    <xs:attribute name="code_491" type="xs:string" use="optional" />
    <xs:attribute name="code_493" type="xs:string" use="optional" />
    <xs:attribute name="code_500" type="xs:string" use="optional" />
    <xs:attribute name="code_501" type="xs:string" use="optional" />
    <xs:attribute name="code_502" type="xs:string" use="optional" />
    <xs:attribute name="code_503" type="xs:string" use="optional" />
    <xs:attribute name="code_504" type="xs:string" use="optional" />
    <xs:attribute name="code_505" type="xs:string" use="optional" />
    <xs:attribute name="code_513" type="xs:string" use="optional" />
    <xs:attribute name="code_600" type="xs:string" use="optional" />
    <xs:attribute name="code_603" type="xs:string" use="optional" />
    <xs:attribute name="code_604" type="xs:string" use="optional" />
    <xs:attribute name="code_606" type="xs:string" use="optional" />
    <xs:attribute name="external" type="xs:string" use="optional" />
  </xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- resultType-->
<xs:complexType name="resultType">
  <xs:all>
    <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
  </xs:all>
</xs:complexType>

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_status_messages_set

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_sip_status_messages_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="sipStatusMessagesOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="sipStatusMessagesOptionsType" external="666:language:en,text:New
desc;language:ru,text:Новое описание|667:language:en,text:Desc"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sip_status_messages_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="external"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Hc_sip_telephone_reboot - Перезагрузка телефонных аппаратов

- [Hc_sip_telephone_reboot](#)

Нс_sip_telephone_reboot

Команда перезагрузки телефонных аппаратов с помощью отправки sip сообщения (NOTIFY).

Аналог команды в CoCon:

`domain/<DOMAIN>/sip/.telephone-reboot`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_telephone_reboot

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- SipUser -->
  <xs:complexType name="SipUser">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="ip_addr" type="xs:string" use="required"/>
    <xs:attribute name="vendor" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:attribute name="code" type="xs:string" use="required"/>
    <xs:attribute name="endpoint" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="code" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
    <xs:attribute name="endpoint" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="sip" type="SipUser" minOccurs="0"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/sip_telephone_reboot

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="sip_telephone_reboot.xsd">
  <sip domain="test.domain" ip_addr="192.168.1.26" vendor="yealink"/>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sip_telephone_reboot.xsd">
  code="200"endpoint="192.168.1.26:10105(udp)"
</out>
```

Нс_sip_timers - Управление таймерами кластера SIP

- [Нс_sip_timers_info](#)
- [Нс_sip_timers_set](#)
- [Нс_sip_timers_clean](#)

Нс_sip_timers_info

Команда для просмотра значений таймеров подсистемы SIP.

Аналог команды в CoCon:

domain/<DOMAIN>/timers/sip/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_timers_info

Код HTTP-ответа::

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show timer properties of SIP cluster.
      Usage: http://server:9999/commands/sip_timers_info.
      Implementation: hc_sip_timers_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- sipTimersOptionsType-->
  <xs:complexType name="sipTimersOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="sip_T1" type="integerType" use="optional"/>
        <xs:attribute name="sip_T2" type="integerType" use="optional"/>
        <xs:attribute name="sip_T4" type="integerType" use="optional"/>
        <xs:attribute name="sip_TC" type="integerType" use="optional"/>
        <xs:attribute name="forbidden" type="integerType" use="optional"/>
        <xs:attribute name="find_me" type="integerType" use="optional"/>
        <xs:attribute name="subscribe_limit" type="integerType" use="optional"/>
        <xs:attribute name="rfc4028_min_se" type="integerType" use="optional"/>
        <xs:attribute name="rfc4028_use_se" type="integerType" use="optional"/>
        <xs:attribute name="subscribe_over" type="integerType" use="optional"/>
        <xs:attribute name="reg_expire_min" type="integerType" use="optional"/>
        <xs:attribute name="reg_expire_max" type="integerType" use="optional"/>
        <xs:attribute name="invite_transaction_timeout" type="integerType" use="optional"
      />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>

```

```

        <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1"/>
        </xs:all>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/sip_timers_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="sip_timers_info.xsd">
  <request>
    <domain_options domain="biysk.local" />
  </request>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_timers_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="sipTimersOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="sipTimersOptionsType" subscribe_over="5" subscribe_limit="0"
        sip_TC="181000" sip_T4="5000" sip_T2="4000" sip_T1="500" rfc4028_use_se="1800"
        rfc4028_min_se="300" reg_expire_min="90" reg_expire_max="3600" invite_transaction_timeout="32000"
        forbidden="100" find_me="10000"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Нс_sip_timers_set

Команда установки значений таймеров подсистемы SIP в миллисекундах.

Аналог команды в CoCon:

domain/<DOMAIN>/timers/sip/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_timers_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set timers on SIP.
      Usage: http://server:9999/commands/sip_timers_set.
      Implementation: hc_sip_timers_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- sipTimersOptionsType-->
  <xs:complexType name="sipTimersOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="sip_T1" type="integerType" use="optional"/>
        <xs:attribute name="sip_T2" type="integerType" use="optional"/>
        <xs:attribute name="sip_T4" type="integerType" use="optional"/>
        <xs:attribute name="sip_TC" type="integerType" use="optional"/>
        <xs:attribute name="forbidden" type="integerType" use="optional"/>
        <xs:attribute name="find_me" type="integerType" use="optional"/>
        <xs:attribute name="subscribe_limit" type="integerType" use="optional"/>
        <xs:attribute name="rfc4028_min_se" type="integerType" use="optional"/>
        <xs:attribute name="rfc4028_use_se" type="integerType" use="optional"/>
        <xs:attribute name="subscribe_over" type="integerType" use="optional"/>
        <xs:attribute name="reg_expire_min" type="integerType" use="optional"/>
        <xs:attribute name="reg_expire_max" type="integerType" use="optional"/>
        <xs:attribute name="invite_transaction_timeout" type="integerType" use="optional"
      />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

        </xs:all>
        </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
            <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_timers_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_timers_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="sipTimersOptionsType" />
      </system>
      <domain name="biysk.local">
        <options xs:type="sipTimersOptionsType" sip_T1="500" sip_T2="600" sip_T4="700"
        forbidden="400" find_me="500" rfc4028_min_se="100000" rfc4028_use_se="190000" subscribe_over="6"
        />
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_timers_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="find_me"/>
        <success option="forbidden"/>
        <success option="rfc4028_min_se"/>
        <success option="rfc4028_use_se"/>
        <success option="sip_T1"/>
        <success option="sip_T2"/>
        <success option="sip_T4"/>
        <success option="subscribe_over"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_sip_timers_clean

Командой устанавливаются значения по умолчанию для таймеров подсистемы SIP в миллисекундах.

Аналог команды в CoSoc:

domain/<DOMAIN>/timers/sip/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_timers_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean timer properties on SIP.
      Usage: http://server:9999/commands/sip_timers_clean.
      Implementation: hc_sip_timers_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_timers_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_timers_clean.xsd">
  <request>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <option name="sip_T1"/>
        <option name="sip_T2"/>
        <option name="sip_T4"/>
        <option name="forbidden"/>
        <option name="find_me"/>
        <option name="session_limit"/>
        <option name="rfc4028_min_se"/>
        <option name="subscribe_over"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_timers_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="find_me"/>
        <success option="forbidden"/>
        <success option="rfc4028_min_se"/>
        <success option="sip_T1"/>
        <success option="sip_T2"/>
        <success option="sip_T4"/>
        <success option="subscribe_over"/>
        <error option="session_limit" reason="{error,not_found}"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Hc_sip_transport - Управление транспортными свойствами кластера SIP

- [Hc_sip_transport_info](#)
- [Hc_sip_transport_set](#)
- [Hc_sip_transport_clean](#)

[Hc_sip_transport_info](#)

Команда позволяет просмотреть свойства sip-transport кластера SIP.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/network/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_transport_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show sip-transport properties of SIP cluster.
      Usage: http://server:9999/commands/sip_transport_info.
      Implementation: hc_sip_transport_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- sipTransportOptionsType-->
  <xs:complexType name="sipTransportOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="ip_set" type="stringType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Usage: ip_set1,ip_set_2,...,ip_set_N
              The list of IP set's separated by comma.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_transport_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="sip_transport_info.xsd">
  <request>
    <domain_options domain="biysk.local" />
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_transport_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="sipTransportOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="sipTransportOptionsType" ip_set="ipset1"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_sip_transport_set

Команда позволяет редактировать свойства sip-transport кластера SIP.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/network/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_transport_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set sip-transport properties of SIP cluster.
      Usage: http://server:9999/commands/sip_transport_set.
      Implementation: hc_sip_transport_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- sipTransportOptionsType-->
  <xs:complexType name="sipTransportOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="ip_set" type="stringType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Usage: ip_set1,ip_set_2,...,ip_set_N
              The list of IP set's separated by comma.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_transport_set

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sip_transport_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="sipTransportOptionsType" />
      </system>
      <domain name="ct.office">
        <options xs:type="sipTransportOptionsType" ip_set="ipset1,ipset2" />
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_transport_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="ct.office">
        <success option="ip_set"/>
      </domain>
    </domain_options>
  </result>
</out>

```


Http_sip_transport_clean

Команда устанавливает значения по умолчанию для свойств sip-transport кластера SIP.

Аналог команды в CoSoc:

`domain/<DOMAIN>/sip/network/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/sip_transport_clean`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean sip-transport properties of SIP cluster.
      Usage: http://server:9999/commands/sip_transport_clean.
      Implementation: hc_sip_transport_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sip_transport_clean

```
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sip_transport_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="ct.office">
        <option name="ip_set" />
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_transport_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="ct.office">
        <success option="ip_set"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Нс_sip_trunk - Управление SIP-транками (HTTP-терминал)

- [Нс_sip_trunk_declare](#)
- [Нс_sip_trunk_set](#)
- [Нс_sip_trunk_show](#)
- [Нс_sip_trunk_remove](#)
- [Нс_sip_trunk_check](#)

Нс_sip_trunk_declare

Команда позволяет создать SIP-транк.

Аналог команды в CoCon:

domain/<DOMAIN>/trunk/sip/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_trunk_declare

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- nonEmptyString -->
  <xs:simpleType name="nonEmptyString">
    <xs:restriction base="xs:string">
      <xs:minLength value="1"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="modeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="sip-proxy" />
      <xs:enumeration value="sip-t" />
      <xs:enumeration value="sip-i" />
      <xs:enumeration value="sip-q" />
    </xs:restriction>
  </xs:simpleType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="sip" minOccurs="0" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="domain" type="nonEmptyString" use="required" />
            <xs:attribute name="context" type="xs:string" use="required" />
            <xs:attribute name="modifier" type="nonEmptyString" use="optional" />
            <xs:attribute name="group" type="nonEmptyString" use="required" />
            <xs:attribute name="iface" type="nonEmptyString" use="required" />
            <xs:attribute name="ip" type="xs:string" use="optional" />
            <xs:attribute name="ip_set" type="xs:string" use="required" />
            <xs:attribute name="port" type="xs:integer" use="optional" />
            <xs:attribute name="listen_port" type="xs:string" use="optional" />
            <xs:attribute name="mode" type="modeType" use="optional" />
            <xs:attribute name="registration" type="xs:boolean" use="optional" />
            <xs:attribute name="force" type="xs:boolean" use="optional" default="true" />
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="progress">
    <xs:complexType>
      <xs:all>
        <xs:element name="text" minOccurs="1" type="valueType" />
        <xs:element name="ref" minOccurs="1" type="valueType" />
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

        </xs:all>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Запрос: http://192.168.1.21:9999/commands/sip_trunk_declare

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_trunk_declare.xsd">
  <sip domain="biysk.local" force="true" listen_port="5075" port="5075" iface="ems5"
  ip_set="ipset1" mode="sip-proxy" context="ctx_from_local" group="ssw.gr" ip="192.168.2.15"/>
</in>

```

Ответ: 201

Нс_sip_trunk_set

Команда позволяет установить или изменить параметры SIP-транка.

Аналог команды в CoCon:

domain/<DOMAIN>/trunk/sip/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_trunk_set

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями::

- 100 - в случае прогресса;
- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- valueType -->
  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="property" type="xs:string" use="required"/>
    <xs:attribute name="message" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- propertyKeyType -->
  <xs:simpleType name="propertyKeyType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="block"/>
      <xs:enumeration value="alarm_enable"/>
      <xs:enumeration value="call_limit"/>
      <xs:enumeration value="reg_expire_min"/>
      <xs:enumeration value="reg_expire_max"/>
      <xs:enumeration value="content_encoding">
        <xs:annotation>
          <xs:documentation>
            Values: gzip|none
          </xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="ip_set"/>
      <xs:enumeration value="ip"/>
      <xs:enumeration value="isup_r_ccl_send"/>
      <xs:enumeration value="listen_port"/>
      <xs:enumeration value="cause_location"/>
      <xs:enumeration value="maddr"/>
      <xs:enumeration value="mode">
        <xs:annotation>
          <xs:documentation>
            Values: sip-t|sip-i|sip-q|sip-proxy
          </xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="my_from"/>
      <xs:enumeration value="options_control"/>
      <xs:enumeration value="port"/>
      <xs:enumeration value="host"/>
      <xs:enumeration value="disable_remote_control"/>
      <xs:enumeration value="inc_authentication"/>
      <xs:enumeration value="inc_authentication_type">
        <xs:annotation>
          <xs:documentation>
            Values: proxy|user
          </xs:documentation>
        </xs:annotation>
      </xs:enumeration>
    </xs:restriction>
  </xs:simpleType>

```

```

        </xs:documentation>
    </xs:annotation>
</xs:enumeration>
<xs:enumeration value="inc_login"/>
<xs:enumeration value="inc_password"/>
<xs:enumeration value="out_login"/>
<xs:enumeration value="out_password"/>
<xs:enumeration value="rel_to_cancel"/>
<xs:enumeration value="req100rel"/>
<xs:enumeration value="rfc4028_control"/>
<xs:enumeration value="context"/>
<xs:enumeration value="modifier"/>
<xs:enumeration value="sip_transport">
    <xs:annotation>
        <xs:documentation>
            Values: tcp_only|udp_only|tcp_prefer|udp_prefer
        </xs:documentation>
    </xs:annotation>
</xs:enumeration>
<xs:enumeration value="sipdomain"/>
<xs:enumeration value="symbol_hash_as_is"/>
<xs:enumeration value="trunk"/>
<xs:enumeration value="remote_network_type">
    <xs:annotation>
        <xs:documentation>
            Values: private|local|transit|international
        </xs:documentation>
    </xs:annotation>
</xs:enumeration>
<xs:enumeration value="allow_user_registration"/>
<xs:enumeration value="dtmf_relay"/>
<xs:enumeration value="dtmf_duration"/>
<xs:enumeration value="referred_by_as_cgpn"/>
<xs:enumeration value="route_by_via"/>
<xs:enumeration value="nat_traversal"/>
<xs:enumeration value="tel_uri_using"/>
<xs:enumeration value="user_name"/>
<xs:enumeration value="out_registration"/>
<xs:enumeration value="area"/>
<xs:enumeration value="compact_form"/>
<xs:enumeration value="transit"/>
<xs:enumeration value="isup_standard">
    <xs:annotation>
        <xs:documentation>
            Values: etsi|rus|ansi|telcordia
        </xs:documentation>
    </xs:annotation>
</xs:enumeration>
<xs:enumeration value="display_encoding">
    <xs:annotation>

```



```

        <xs:documentation>
            Values: default|win1251
        </xs:documentation>
    </xs:annotation>
</xs:enumeration>
<xs:enumeration value="category_to_sip"/>
<xs:enumeration value="invite_transaction_timeout"/>
</xs:restriction>
</xs:simpleType>
<!-- propertyType -->
<xs:complexType name="propertyType">
    <xs:attribute name="name" type="propertyKeyType" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
</xs:complexType>
<!-- sipType -->
<xs:complexType name="sipType">
    <xs:all>
        <xs:element name="properties">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:all>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="group" type="xs:string" use="required"/>
    <xs:attribute name="iface" type="xs:string" use="optional"/>
    <xs:attribute name="force" type="xs:boolean" use="optional" default="true"/>

```

```

        <xs:element name="error" type="okType" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:all>
</xs:complexType>
</xs:element>
<!-- progress -->
<xs:element name="progress">
  <xs:complexType>
    <xs:all>
      <xs:element name="text" type="valueType"/>
      <xs:element name="ref" type="valueType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/sip_trunk_set

```

<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="sip_trunk_set.xsd">
  <sip group="ssw.gr" iface="ems5" domain="biysk.local">
    <properties>
      <property name="alarm_enable" value="true"/>
      <property name="options_control" value="60"/>
      <property name="mode" value="sip-t"/>
    </properties>
  </sip>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_trunk_set.xsd">
  <response>
    <ok property="alarm_enable"/>
    <ok property="options_control"/>
    <ok property="mode"/>
  </response>
</out>

```

Может вернуть прогресс:

```
<?xml version="1.0"?>
<progress xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sip_trunk_set.xsd">
  <ref value="{md@sky, {1314, 939007, 922379}}"/>
  <state value="set_result"/>
  <ip value="127.0.0.1"/>
  <port value="1234"/>
</progress>
```

[Hc_sip_trunk_show](#)

Команда для просмотра настроек SIP-транка.

Аналог команды в CoCon:

domain/<DOMAIN>/trunk/sip/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_trunk_show

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить двумя сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- rfc4028ControlType -->
  <xs:simpleType name="rfc4028ControlType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="true" />
      <xs:enumeration value="false" />
      <xs:enumeration value="force" />
    </xs:restriction>
  </xs:simpleType>
  <!-- displayEncodingType -->
  <xs:simpleType name="displayEncodingType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="default"/>
      <xs:enumeration value="win1251"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="groupType">
    <xs:attribute name="group" type="xs:string" use="required" />
    <xs:attribute name="call_count" type="xs:integer" use="required" />
    <xs:attribute name="my_from" type="xs:string" use="optional" />
    <xs:attribute name="context" type="xs:string" use="optional" />
    <xs:attribute name="modificator" type="xs:string" use="optional" />
  </xs:complexType>
  <xs:complexType name="iface_shortType">
    <xs:attribute name="uri" type="xs:string" use="required" />
    <xs:attribute name="group" type="xs:string" use="required" />
    <xs:attribute name="isActive" type="xs:boolean" use="required" />
    <xs:attribute name="mode" type="xs:string" use="required" />
    <xs:attribute name="dynamic" type="xs:string" use="required" />
    <xs:attribute name="block" type="xs:boolean" use="required" />
    <xs:attribute name="host" type="xs:string" use="optional" />
    <xs:attribute name="port" type="xs:integer" use="optional" />
    <xs:attribute name="ip_set" type="xs:string" use="optional" />
    <xs:attribute name="listen_port" type="xs:string" use="optional" />
    <xs:attribute name="context" type="xs:string" use="optional" />
    <xs:attribute name="modificator" type="xs:string" use="optional" />
  </xs:complexType>
  <xs:complexType name="ifaceType">
    <xs:attribute name="uri" type="xs:string" use="required" />
    <xs:attribute name="group" type="xs:string" use="required" />
    <xs:attribute name="isActive" type="xs:boolean" use="required" />
    <xs:attribute name="mode" type="xs:string" use="required" />
    <xs:attribute name="dynamic" type="xs:string" use="required" />
    <xs:attribute name="block" type="xs:boolean" use="required" />
    <!-- optional -->
    <xs:attribute name="my_from" type="xs:string" use="optional" />
    <xs:attribute name="context" type="xs:string" use="optional" />
    <xs:attribute name="modificator" type="xs:string" use="optional" />
  </xs:complexType>

```

```

<xs:attribute name="alarm_enable" type="xs:boolean" use="optional" />
<xs:attribute name="host" type="xs:string" use="optional" />
<xs:attribute name="port" type="xs:integer" use="optional" />
<xs:attribute name="ip_set" type="xs:string" use="optional" />
<xs:attribute name="listen_port" type="xs:string" use="optional" />
<xs:attribute name="auth_type" type="xs:string" use="optional" />
<xs:attribute name="rel_to_cancel" type="xs:boolean" use="optional" />
<xs:attribute name="isup_r_ccl_send" type="xs:boolean" use="optional" />
<xs:attribute name="req100rel" type="xs:boolean" use="optional" />
<xs:attribute name="symbol_hash_as_is" type="xs:boolean" use="optional" />
<xs:attribute name="disable_remote_control" type="xs:boolean" use="optional" />
<xs:attribute name="inc_authentication" type="xs:boolean" use="optional" />
<xs:attribute name="rfc4028_control" type="rfc4028ControlType" use="optional" />
<xs:attribute name="cause_location" type="xs:string" use="optional" />
<xs:attribute name="inc_authentication_type" type="xs:string" use="optional" />
<xs:attribute name="content_encoding" type="xs:string" use="optional" />
<xs:attribute name="sip_transport" type="xs:string" use="optional" />
<xs:attribute name="sip_i_support" type="xs:boolean" use="optional" />
<xs:attribute name="options_control" type="xs:integer" use="optional" />
<xs:attribute name="call_count" type="xs:integer" use="optional" />
<xs:attribute name="reg_expire_min" type="xs:integer" use="optional" />
<xs:attribute name="reg_expire_max" type="xs:integer" use="optional" />
<xs:attribute name="trunk" type="xs:string" use="optional" />
<xs:attribute name="inc_login" type="xs:string" use="optional" />
<xs:attribute name="inc_password" type="xs:string" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      In case of strong security restrictions password may be shown as *****
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="out_login" type="xs:string" use="optional" />
<xs:attribute name="out_password" type="xs:string" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      In case of strong security restrictions password may be shown as *****
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="maddr" type="xs:string" use="optional" />
<xs:attribute name="sipdomain" type="xs:string" use="optional" />
<xs:attribute name="iface" type="xs:string" use="optional" />
<xs:attribute name="display_encoding" type="displayEncodingType" use="optional"/>
<xs:attribute name="category_to_sip" type="xs:string" use="optional"/>
<xs:attribute name="invite_transaction_timeout" type="xs:integer" use="optional"/>
</xs:complexType>

<xs:element name="in">
  <xs:complexType>
    <xs:all>

```

```

        <xs:element name="sip" minOccurs="0" maxOccurs="1">
            <xs:complexType>
                <xs:attribute name="domain" type="xs:string" use="required" />
                <xs:attribute name="group" type="xs:string" use="required" />
                <xs:attribute name="id" type="xs:string" use="optional" />
            </xs:complexType>
        </xs:element>
    </xs:all>
</xs:complexType>
</xs:element>

<xs:element name="out">
    <xs:complexType>
        <xs:choice minOccurs="0">
            <xs:element name="group_info" type="groupType" maxOccurs="unbounded"/>
            <xs:element name="iface_info" type="ifaceType" maxOccurs="unbounded"/>
            <xs:element name="iface_short_info" type="iface_shortType" maxOccurs="unbounded"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Запрос: http://192.168.1.21:9999/commands/sip_trunk_show

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_trunk_show.xsd">
  <sip domain="biysk.local" group="ssw.gr" id="ems2" />
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_trunk_show.xsd">
  <iface_info invite_transaction_timeout="32000" display_encoding="default"
  context="ctx_from_local" group="ssw.gr" rfc4028_control="force" disable_remote_control="false"
  symbol_hash_as_is="false" trunk="biysk.local.ssw.gr.trunk.autoname" alarm_enable="false"
  my_from="biysk.local" category_to_sip="default" sip_transport="udp_only"
  cause_location="default" rel_to_cancel="false" maddr="none" content_encoding="none"
  out_password="*****" out_login="none" inc_password="*****" inc_login="none"
  inc_authentication_type="user" inc_authentication="true" options_control="0" block="false"
  listen_port="5072" ip_set="ipset1" port="5072" host="192.168.2.12" mode="sip-proxy"
  dynamic="false" sipdomain="use-address" req100rel="false" uri="ems2" isActive="true"/>
</out>

```

Нс_sip_trunk_remove

Команда позволяет удалить SIP-транк.

Аналог команды в CoCon:

`domain/<DOMAIN>/trunk/sip/remove`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_trunk_remove

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="sip" minOccurs="0" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required" />
            <xs:attribute name="group" type="xs:string" use="required" />
            <xs:attribute name="force" type="xs:boolean" use="optional" />
            <xs:attribute name="id" type="xs:string" use="optional" />
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/commands/sip_trunk_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_trunk_remove.xsd">
  <sip domain="biysk.local" group="ssw.gr" id="ems5" />
</in>
```

Ответ с требованием подтверждения:

status: 300

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="modal.xsd">
  <modal ref="md1@ecss1:1475" from="remove" passwd="false" results="yesno">
    <header template="Remove interfaces. Undo will be impossible"/>
    <prompt template="continue"/>
    <error template="You should enter 'yes' or 'no'"/>
  </modal>
</out>
```

Запрос безусловного удаления:

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_trunk_remove.xsd">
  <sip domain="biysk.local" group="ssw.gr" id="ems5" force="true" />
</in>
```

Ответ: 204

[Hc_sip_trunk_check](#)

Команда для проверки доступности SIP-транка.

Аналог команды в CoSop:

domain/<DOMAIN>/trunk/sip/check

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_trunk_check

Код HTTP-ответа::

- 204 – в случае успеха;

- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- trunkType -->
  <xs:complexType name="trunkType">
    <xs:attribute name="accessible" type="xs:boolean" use="required"/>
    <xs:attribute name="iface" type="xs:string" use="required"/>
    <xs:attribute name="info" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- trunksType -->
  <xs:complexType name="trunksType">
    <xs:sequence>
      <xs:element name="trunk" type="trunkType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="trunk" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="group" type="xs:string" use="required"/>
          <xs:attribute name="iface" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:all>
      <xs:element name="trunks" type="trunksType" minOccurs="0" maxOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType" minOccurs="0" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:Запрос: http://192.168.1.21:9999/commands/sip_trunk_check

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_trunk_check.xsd">
  <request domain="biysk.local">
    <trunk group="smg.gr" iface="smg-4" />
    <trunk group="sbc.gr" iface="to_aster" />
    <trunk group="ssw.gr" iface="ems1" />
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sip_trunk_check.xsd">
  <response>
    <trunks>
      <trunk iface="smg-4" accessible="true" info="OPTIONS to <sip:192.168.2.8>: 200 OK,
cseq: 83004 OPTIONS. Request by node: sip1@ecss1"/>
      <trunk iface="to_aster" accessible="false" info="Response timeout is expired. Request
by node: sip1@ecss1"/>
      <trunk iface="ems1" accessible="false" info="Response timeout is expired. Request by
node: sip1@ecss1"/>
    </trunks>
  </response>
</out>
```

Hc_sip_user - Управление SIP-абонентами

- [Hc_sip_user_declare](#)
- [Hc_sip_user_show](#)
- [Hc_sip_user_registered](#)
- [Hc_sip_user_set](#)
- [Hc_sip_user_remove](#)
- [Hc_sip_user_check](#)

Hc_sip_user_declare

Команда позволяет создать SIP-абонента.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/user/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_user_declare

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 206 - в случае прогресса;
- 300 - в случае модала;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd" />

  <!-- causeType-->
  <xs:simpleType name="causeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="already_exists"/>
      <xs:enumeration value="licence_limit"/>
      <xs:enumeration value="domain_limit"/>
      <xs:enumeration value="no_such_iface"/>
      <xs:enumeration value="bad_owner"/>
      <xs:enumeration value="bad_group"/>
      <xs:enumeration value="cannot_get_iface"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- warningType-->
  <xs:complexType name="warningType">
    <xs:attribute name="cause" type="causeType"/>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="interface" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- aliasType-->
  <xs:complexType name="aliasType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="interface" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
  </xs:complexType>

  <!-- successType-->
  <xs:complexType name="successType">
    <xs:sequence>
      <xs:element name="alias" type="aliasType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- warningsType-->
  <xs:complexType name="warningsType">
    <xs:sequence>
      <xs:element name="warning" type="warningType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- resultType-->
  <xs:complexType name="resultType">

```

```

    <xs:sequence>
      <xs:element name="success" type="successType"/>
      <xs:element name="warnings" type="warningsType"/>
    </xs:sequence>
  </xs:complexType>

  <xs:simpleType name = "yesnoType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="yes" />
      <xs:enumeration value="no" />
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:simpleType name="nonEmptyString">
    <xs:restriction base="xs:string">
      <xs:minLength value="1"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" minOccurs="0" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="domain" type="nonEmptyString" use="required" />
            <xs:attribute name="context" type="nonEmptyString" use="required" />
            <xs:attribute name="modifier" type="nonEmptyString" use="optional" />
            <xs:attribute name="group" type="nonEmptyString" use="required" />
            <xs:attribute name="iface" type="nonEmptyString" use="required" />
            <xs:attribute name="force" type="xs:boolean" use="optional" />
            <xs:attribute name="auth_qop" type="yesnoType" use="optional" />
            <xs:attribute name="login" type="xs:string" use="optional" /> <!--
login_as_number -->
            <xs:attribute name="password" type="xs:string" use="optional" /> <!--
auto_generation -->
            <xs:attribute name="address" type="xs:string" use="optional" />
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="progress">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="ref" minOccurs="1" maxOccurs="1" type="valueType" />

```

```

        <xs:element name="text" minOccurs="1" maxOccurs="1" type="valueType" />
    </xs:sequence>
</xs:complexType>
</xs:element>

<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result" type="resultType" />
            <xs:element name="error" type="errorType">
                <xs:annotation>
                    <xs:documentation xml:lang="en">
                        Available error reasons:
                        - name_is_busy
                        - context_not_found
                        - alias_declaration_error
                        - context_validation_error
                        - invalid_user_name
                        - invalid_alias_digits
                        - cancelled
                    </xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:choice>
    </xs:complexType>
</xs:element>

</xs:schema>

```

Если password равен auto-generation, то пароль будет сгенерирован автоматически.

Пример:

Запрос: http://192.168.1.21:9999/commands/sip_user_declare

```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local" context="ctx_from_local" group="loc.gr" iface="240901" address="
alias-as-user" auth_qop="yes" login="240901" password="auto-generation"/>
</in>

```

Ответ: 206

```
<?xml version="1.0"?>
<progress
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sip_user_d
eclare.xsd">
  <ref value="md1@ecss1:1218"/>
  <text value="Declaration for range: 240901@biysk.local..240901@biysk.local (1)"/>
</progress>
```

В некоторых случаях при декларации может прийти код ответа 300 и xml с описанием модала, например, при декларации уже существующего абонента.

Также может слать код ответа 300 и xml с описанием прогресса.

[Hc_sip_user_show](#)

Команда для просмотра настроек SIP-абонента. Позволяет просматривать всю информацию об абоненте (complete="true"), краткую информацию об абоненте (complete="false"), аутентификационную информацию об абоненте (auth="true"). Если аутентификационной информации нет, поля "login", "password" не заполняются.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/user/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_user_show

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- rfc4028ControlType -->
  <xs:simpleType name="rfc4028ControlType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="true" />
      <xs:enumeration value="false" />
      <xs:enumeration value="force" />
    </xs:restriction>
  </xs:simpleType>
  <!-- authTypeType -->
  <xs:simpleType name="authTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="none"/>
      <xs:enumeration value="always"/>
      <xs:enumeration value="register"/>
      <xs:enumeration value="ldap"/>
      <xs:enumeration value="ds"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- displayEncodingType -->
  <xs:simpleType name="displayEncodingType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="default"/>
      <xs:enumeration value="win1251"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- remoteCtrIndicationType -->
  <xs:simpleType name="remoteCtrIndicationType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="pai"/>
      <xs:enumeration value="rpi"/>
      <xs:enumeration value="none"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- userType -->
  <xs:complexType name="userType">
    <xs:attribute name="active" type="xs:boolean" use="required"/>
    <xs:attribute name="alarm_enable" type="xs:boolean" use="optional"/>
    <xs:attribute name="aliases_list" type="xs:string" use="optional"/>
    <xs:attribute name="auth" type="authTypeType" use="optional"/>
    <xs:attribute name="auth_qop" type="xs:string" use="optional"/>
    <xs:attribute name="digest" type="xs:boolean" use="optional"/>
    <xs:attribute name="contact" type="xs:string" use="required"/>
    <xs:attribute name="domain" type="xs:string" use="optional"/>
    <xs:attribute name="fork_mode" type="xs:string" use="optional"/>
    <xs:attribute name="fixed_contact" type="xs:string" use="optional"/>
    <xs:attribute name="id" type="xs:string" use="optional"/>
    <xs:attribute name="group" type="xs:string" use="required"/>
  </xs:complexType>

```

```

<xs:attribute name="iface" type="xs:string" use="required"/>
<xs:attribute name="login" type="xs:string" use="optional"/>
<xs:attribute name="last_via" type="xs:string" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Format:PROTO_1 HOST_1 PARAM_11=VALUE_11, PARAM_22=VALUE_22,...;PROTO_2 HOST_2
PARAM_21=VALUE_21, PARAM_22=VALUE_22,...;...
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="nat_traversal" type="xs:boolean" use="optional"/>
<xs:attribute name="owner" type="xs:string" use="optional"/>
<xs:attribute name="password" type="xs:string" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      In case of strong security restrictions password may be shown as *****
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attribute name="referred_by_as_cgpn" type="xs:boolean" use="optional"/>
<xs:attribute name="reg_expire_min" type="xs:integer" use="optional"/>
<xs:attribute name="reg_expire_max" type="xs:integer" use="optional"/>
<xs:attribute name="rfc4028_control" type="rfc4028ControlType" use="optional"/>
<xs:attribute name="routing_context" type="xs:string" use="optional"/>
<xs:attribute name="modificator" type="xs:string" use="optional"/>
<xs:attribute name="symbol_hash_as_is" type="xs:boolean" use="optional"/>

<xs:attribute name="tel_uri_in_diversion" type="xs:boolean" use="optional"/>
<xs:attribute name="tel_uri_using" type="xs:boolean" use="optional"/>
<xs:attribute name="sip_domain" type="xs:string" use="optional"/>
<xs:attribute name="sip_modifications" type="xs:string" use="optional"/>
<xs:attribute name="req100rel" type="xs:boolean" use="optional"/>
<xs:attribute name="dtmf_relay" type="xs:boolean" use="optional"/>
<xs:attribute name="original_cdpn_to" type="xs:boolean" use="optional"/>
<xs:attribute name="my_from" type="xs:string" use="optional"/>
<xs:attribute name="transit" type="xs:string" use="optional"/>
<xs:attribute name="auto_answer_version" type="xs:string" use="optional"/>
<xs:attribute name="display_encoding" type="displayEncodingType" use="optional"/>
<xs:attribute name="remote_ctr_indication" type="remoteCtrIndicationType" use="optional"/>
>
<xs:attribute name="category_to_sip" type="xs:string" use="optional"/>
<xs:attribute name="registered_domain_to_invite" type="xs:boolean" use="optional"/>

<xs:attribute name="trusted_ip" type="xs:string" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Format:STRICT;IP:PORT;IP1-IP2:PORT_1-PORT2;IP3;...
      Sample:false;192.168.23.87:9988;192.168.23.181:9900-9910;192.168.23.0
    </xs:documentation>
  </xs:annotation>

```

```

    </xs:attribute>
    <xs:attribute name="user_agent" type="xs:string" use="optional"/>
</xs:complexType>
<!-- usersType -->
<xs:complexType name="usersType">
  <xs:sequence>
    <xs:element name="user" type="userType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="sip" minOccurs="0">
        <xs:complexType>
          <xs:attribute name="domain" type="xs:string" use="required"/>
          <xs:attribute name="group" type="xs:string" use="required"/>
          <xs:attribute name="id" type="xs:string" use="optional"/>
          <xs:attribute name="complete" type="xs:boolean" use="optional"/>
          <xs:attribute name="auth" type="xs:boolean" use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="users" type="usersType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Запрос: http://192.168.1.21:9999/commands/sip_user_show

Показать авторизационные данные всех SIP абонентов в группе loc.gr.

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="sip_user
_show.xsd">
  <sip group="loc.gr" domain="biysk.local" complete="false" auth="true"/>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sip_user_s
how.xsd">
  <users>
    <user active="true" auth="ds" contact="q: 1.0; ip-set: ipset1; on port: 5060; sip <sip:
240101@192.168.2.26:5060;transport=UDP;line=20457>;expires=459" group="loc.gr"
id="063015220171e7f3" iface="240101@biysk.local" login="240101" password="*****"
user_agent="user-agent: Yealink SIP-T28P 2.73.14.3"/>
    <user active="true" auth="ds" contact="q: 1.0; ip-set: ipset1; on port: 5060; sip <sip:
240699@192.168.2.26:5060;transport=UDP;line=42859>;expires=301" group="loc.gr"
id="06301521c46e49a6" iface="240699@biysk.local" login="240699" password="*****"
user_agent="user-agent: Asterisk PBX 13.1.0-dfsg-1.1ubuntu4.1"/>
    <user active="true" auth="ds" contact="q: 1.0; ip-set: ipset1; on port: 5060; sip <sip:
240462@192.168.2.26:5060;transport=UDP;line=18472>;expires=110" group="loc.gr"
id="06301521a9200f53" iface="240462@biysk.local" login="240462" password="*****"
user_agent="user-agent: Brcm-Callctrl/v1.10.3 M5T SIP Stack/4.1.2.2"/>
    <user active="true" auth="ds" contact="q: 1.0; ip-set: ipset1; on port: 5060; sip <sip:
240244@192.168.2.26:5060;transport=UDP;line=45980>;expires=892" group="loc.gr"
id="06301521a7a2c0aa" iface="240244@biysk.local" login="240244" password="*****"
user_agent="user-agent: Yealink SIP-T28P 2.73.14.3"/>
    ...
    <user active="false" auth="ds" contact="" group="loc.gr" id="06301521a5b333b6" iface="240
100@biysk.local" login="240100" password="*****" user_agent=""/>
    <user active="false" auth="ds" contact="" group="loc.gr" id="06301521a48b1b0e" iface="240
006@biysk.local" login="240006" password="*****" user_agent=""/>
    <user active="false" auth="ds" contact="" group="loc.gr" id="06301521a182c308" iface="240
002@biysk.local" login="240002" password="*****" user_agent=""/>
    <user active="false" auth="ds" contact="" group="loc.gr" id="063015219eda555a" iface="240
001@biysk.local" login="240001" password="*****" user_agent=""/>
  </users>
</out>

```

Пример 2

Запрос: http://192.168.1.21:9999/commands/sip_user_show

Показать все данные об SIP абоненте 240501@biysk.local в группе loc.gr.

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="sip_user
_show.xsd">
  <sip group="loc.gr" id="240501@biysk.local" domain="biysk.local" complete="true"/>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sip_user_s
how.xsd">
  <users>
    <user active="true" alarm_enable="false" aliases_list="240501" auth="ds" auth_qop="true"
auto_answer_version="default" category_to_sip="default" contact="q: 1.0; ip-set: ipset1; on port:
5060; sip <sip:240501@192.168.2.26:5060;transport=UDP;line=42098>;expires=328" digest="true"
display_encoding="default" domain="biysk.local" dtmf_relay="false" fork_mode="all-contacts"
group="loc.gr" id="06301521b463510a" iface="240501@biysk.local" last_via="SIP/2.0/UDP
192.168.2.26 branch=z9hG4bKPjzAaAU5198fMYk7vFLbjDNk4AP.GBS6T, received=192.168.2.26,
rport=5060;SIP/2.0/UDP 10.22.128.19 received=10.22.128.19, branch=z9hG4bK53e6dedd" login="240501"
my_from="biysk.local" nat_traversal="false" original_cdpn_to="true" owner="sip1"
password="*****" referred_by_as_cgpn="false" reg_expire_max="3600" reg_expire_min="90"
registered_domain_to_invite="true" remote_ctr_indication="rpi" req100rel="false"
rfc4028_control="force" routing_context="ctx_from_local" sip_domain="biysk.local"
sip_modifications="" symbol_hash_as_is="false" tel_uri_in_diversion="false" transit=""
trusted_ip="false" user_agent="user-agent: Asterisk PBX 13.1.0~dfsg-1.1ubuntu4.1"/>
  </users>
</out>

```

Нс_sip_user_registered

Команда выводит список зарегистрированных SIP-абонентов.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/user/registered

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_user_registered

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- registrationType -->
  <xs:complexType name="registrationType">
    <xs:attribute name="iface" type="xs:string" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
    <xs:attribute name="dynamic" type="xs:boolean" use="required"/>
    <xs:attribute name="user" type="xs:string" use="required"/>
    <xs:attribute name="sip_domain" type="xs:string" use="required"/>
    <xs:attribute name="reg_on" type="xs:string" use="required"/>
    <xs:attribute name="contact" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- registrationsType -->
  <xs:complexType name="registrationsType">
    <xs:sequence>
      <xs:element name="registration" type="registrationType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="sip" minOccurs="0">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="group" type="xs:string" use="required"/>
            <xs:attribute name="id" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="registrations" type="registrationsType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

Запрос: http://192.168.1.21:9999/commands/sip_user_registered

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sip_user_registered.xsd">
  <sip domain="biysk.local" group="loc.gr"/>
</in>
```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sip_user_r
egistered.xsd">
  <registrations>
    <registration contact="q: 1.0; ip-set: ipset1; on port: 5060; sipregistred by node:
sip1@ecss1sip transport: system (udp_only)user-agent: Twinkle/1.10.1<sip:
240100@192.168.2.14>;expires=555" dynamic="false" reg_on="registered on: node: sip1@ecss1 port:
udp 192.168.2.61:5060 last via list: SIP/2.0/UDP 192.168.2.14;received=192.168.2.14;rport=5060
last user agent: Twinkle/1.10.1 " iface="240100@biysk.local"/>
    <registration contact="q: 1.0; ip-set: ipset1; on port: 5060; sipregistred by node:
sip1@ecss1sip transport: udpuser-agent: Yealink SIP-T28P 2.73.14.3<sip:240101@192.168.2.26:5060;t
ransport=UDP;line=20457>;expires=737" dynamic="false" reg_on="registered on: node: sip1@ecss1
port: udp 192.168.2.62:5060 last via list: SIP/2.0/UDP
192.168.2.26:5060;received=192.168.2.26;rport=5060 SIP/2.0/UDP
192.168.1.201:5062;received=194.150.142.251 last user agent: Yealink SIP-T28P 2.73.14.3 "
iface="240101@biysk.local"/>
    <registration contact="q: 1.0; ip-set: ipset1; on port: 5060; sipregistred by node:
sip1@ecss1sip transport: udpuser-agent: Yealink SIP-T28P 2.73.14.3<sip:240244@192.168.2.26:5060;t
ransport=UDP;line=39810>;expires=734" dynamic="false" reg_on="registered on: node: sip1@ecss1
port: udp 192.168.2.62:5060 last via list: SIP/2.0/UDP
192.168.2.26:5060;received=192.168.2.26;rport=5060 SIP/2.0/UDP
192.168.1.201:5067;received=194.150.142.251 last user agent: Yealink SIP-T28P 2.73.14.3 "
iface="240244@biysk.local"/>
    <registration contact="q: 1.0; ip-set: ipset1; on port: 5060; sipregistred by node:
sip1@ecss1sip transport: udpuser-agent: Brcm-Callctrl/v1.10.3 M5T SIP Stack/4.1.2.2<sip:
240462@192.168.2.26:5060;transport=UDP;line=20445>;expires=573" dynamic="false"
reg_on="registered on: node: sip1@ecss1 port: udp 192.168.2.62:5060 last via list: SIP/2.0/UDP
192.168.2.26:5060;received=192.168.2.26;rport=5060 SIP/2.0/UDP 10.28.32.131;received=10.28.32.131
last user agent: Brcm-Callctrl/v1.10.3 M5T SIP Stack/4.1.2.2 " iface="240462@biysk.local"/>
    <registration contact="q: 1.0; ip-set: ipset1; on port: 5060; sipregistred by node:
sip1@ecss1sip transport: system (udp_only)user-agent: TAU-2M.IP/2.0.0.229 SN/VI39005419 sofia-
sip/1.12.10<sip:240464@192.168.2.200:5064>;expires=937" dynamic="false" reg_on="registered on:
node: sip1@ecss1 port: udp 192.168.2.61:5060 last via list: SIP/2.0/UDP
192.168.2.200:5064;received=192.168.2.200;rport=5064 last user agent: TAU-2M.IP/2.0.0.229 SN/
VI39005419 sofia-sip/1.12.10 " iface="240464@biysk.local"/>
    <registration contact="q: 1.0; ip-set: ipset1; on port: 5060; sipregistred by node:
sip1@ecss1sip transport: system (udp_only)user-agent: TAU-2M.IP/2.0.0.229 SN/VI39005419 sofia-
sip/1.12.10<sip:240465@192.168.2.200:5064>;expires=1242" dynamic="false" reg_on="registered on:
node: sip1@ecss1 port: udp 192.168.2.61:5060 last via list: SIP/2.0/UDP
192.168.2.200:5064;received=192.168.2.200;rport=5064 last user agent: TAU-2M.IP/2.0.0.229 SN/
VI39005419 sofia-sip/1.12.10 " iface="240465@biysk.local"/>
    <registration contact="q: 1.0; ip-set: ipset1; on port: 5060; sipregistred by node:
sip1@ecss1sip transport: udpuser-agent: TAU-8.IP/2.6.2 SN/VI33023289 sofia-sip/1.12.10<sip:
240466@192.168.2.26:5060;transport=UDP;line=26482>;expires=882" dynamic="false"
reg_on="registered on: node: sip1@ecss1 port: udp 192.168.2.62:5060 last via list: SIP/2.0/UDP
192.168.2.26:5060;received=192.168.2.26;rport=5060 SIP/2.0/UDP
10.25.20.62;received=10.25.20.62;rport=5060 last user agent: TAU-8.IP/2.6.2 SN/VI33023289 sofia-
sip/1.12.10 " iface="240466@biysk.local"/>
  
```



```

    <registration contact="q: 1.0; ip-set: ipset1; on port: 5060; sipregistred by node:
sip1@ecss1sip transport: udpuser-agent: Brcm-Callctrl/v1.10.3 M5T SIP Stack/4.1.2.2<sip:
240469@192.168.2.26:5060;transport=UDP;line=58471>;expires=515" dynamic="false"
reg_on="registered on: node: sip1@ecss1 port: udp 192.168.2.62:5060 last via list: SIP/2.0/UDP
192.168.2.26:5060;received=192.168.2.26;rport=5060 SIP/2.0/UDP 10.28.32.131;received=10.28.32.131
last user agent: Brcm-Callctrl/v1.10.3 M5T SIP Stack/4.1.2.2 " iface="240469@biysk.local"/>
    <registration contact="q: 1.0; ip-set: ipset1; on port: 5060; sipregistred by node:
sip1@ecss1sip transport: udpuser-agent: RG-1404GF-W/1.11.0 SN/VI25002012 sofia-sip/1.12.10<sip:
240470@192.168.2.26:5060;transport=UDP;line=1311>;expires=737" dynamic="false" reg_on="registered
on: node: sip1@ecss1 port: udp 192.168.2.62:5060 last via list: SIP/2.0/UDP
192.168.2.26:5060;received=192.168.2.26;rport=5060 SIP/2.0/UDP
10.25.1.23:5069;received=10.25.1.23;rport=5069 last user agent: RG-1404GF-W/1.11.0 SN/VI25002012
sofia-sip/1.12.10 " iface="240470@biysk.local"/>
    <registration contact="q: 1.0; ip-set: ipset1; on port: 5060; sipregistred by node:
sip1@ecss1sip transport: udpuser-agent: RG-1404GF-W/1.11.0 SN/VI25002012 sofia-sip/1.12.10<sip:
240474@192.168.2.26:5060;transport=UDP;line=25653>;expires=1595" dynamic="false"
reg_on="registered on: node: sip1@ecss1 port: udp 192.168.2.62:5060 last via list: SIP/2.0/UDP
192.168.2.26:5060;received=192.168.2.26;rport=5060 SIP/2.0/UDP
10.25.1.23:5074;received=10.25.1.23;rport=5074 last user agent: RG-1404GF-W/1.11.0 SN/VI25002012
sofia-sip/1.12.10 " iface="240474@biysk.local"/>
    <registration contact="q: 1.0; ip-set: ipset1; on port: 5060; sipregistred by node:
sip1@ecss1sip transport: udpuser-agent: Asterisk PBX 13.1.0~dfsg-1.1ubuntu4.1<sip:
240500@192.168.2.26:5060;transport=UDP;line=30684>;expires=360" dynamic="false"
reg_on="registered on: node: sip1@ecss1 port: udp 192.168.2.62:5060 last via list: SIP/2.0/UDP
192.168.2.26:5060;received=192.168.2.26;rport=5060 SIP/2.0/UDP
10.22.128.19:5060;received=10.22.128.19 last user agent: Asterisk PBX 13.1.0~dfsg-1.1ubuntu4.1 "
iface="240500@biysk.local"/>
  </registrations>
</out>

```

Если абонент не зарегистрирован, он не будет показан в списке.

Пример 2:

Запрос: http://192.168.1.21:9999/commands/sip_user_registered

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sip_user_r
egistered.xsd">
  <sip domain="biysk.local" group="loc.gr" id="240466@biysk.local"/>
</in>

```

Ответ:200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sip_user_r
egistered.xsd">
  <registrations>
    <registration contact="q: 1.0; ip-set: ipset1; on port: 5060; sipregistred by node:
sip1@ecss1sip transport: udpuser-agent: TAU-8.IP/2.6.2 SN/VI33023289 sofia-sip/1.12.10<sip:
240466@192.168.2.26:5060;transport=UDP;line=26482>;expires=1805" dynamic="false"
reg_on="registered on: node: sip1@ecss1 port: udp 192.168.2.61:5060 last via list: SIP/2.0/UDP
192.168.2.26:5060;received=192.168.2.26;rport=5060 SIP/2.0/UDP
10.25.20.62;received=10.25.20.62;rport=5060 last user agent: TAU-8.IP/2.6.2 SN/VI33023289 sofia-
sip/1.12.10 " iface="240466@biysk.local"/>
  </registrations>
</out>

```

Нс_sip_user_set

Команда позволяет установить или изменить параметры SIP-абонента.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/user/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_user_set

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- rfc4028ControlType -->
  <xs:simpleType name="rfc4028ControlTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="true" />
      <xs:enumeration value="false" />
      <xs:enumeration value="force" />
    </xs:restriction>
  </xs:simpleType>
  <!-- authTypeType -->
  <xs:simpleType name="authTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="none"/>
      <xs:enumeration value="always"/>
      <xs:enumeration value="register"/>
      <xs:enumeration value="ldap"/>
      <xs:enumeration value="ds"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- displayEncodingValueType -->
  <xs:simpleType name="displayEncodingValueType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="default"/>
      <xs:enumeration value="win1251"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- remoteCtrIndicationValueType -->
  <xs:simpleType name="remoteCtrIndicationValueType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="pai"/>
      <xs:enumeration value="rpi"/>
      <xs:enumeration value="none"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- authType -->
  <xs:complexType name="authType">
    <xs:attribute name="value" type="authTypeType" use="required"/>
  </xs:complexType>
  <!-- rfc4028ControlType -->
  <xs:complexType name="rfc4028ControlType">
    <xs:attribute name="value" type="rfc4028ControlTypeType" use="required"/>
  </xs:complexType>
  <!-- qopType -->
  <xs:complexType name="qopType">
    <xs:attribute name="value" type="xs:boolean" use="required"/>
  </xs:complexType>
  <!-- loginType -->
  <xs:complexType name="loginType">

```

```

    <xs:attribute name="login" type="xs:string" use="required"/>
    <xs:attribute name="password" type="xs:string"/>
</xs:complexType>
<!-- stringType -->
<xs:complexType name="stringType">
    <xs:attribute name="value" type="xs:string" use="required"/>
</xs:complexType>
<!-- integerType -->
<xs:complexType name="integerType">
    <xs:attribute name="value" type="xs:integer" use="required"/>
</xs:complexType>
<!-- booleanType -->
<xs:complexType name="booleanType">
    <xs:attribute name="value" type="xs:boolean" use="required"/>
</xs:complexType>
<!-- displayEncodingType -->
<xs:complexType name="displayEncodingType">
    <xs:attribute name="value" type="displayEncodingValueType" use="required"/>
</xs:complexType>
<!-- displayEncodingType -->
<xs:complexType name="remoteCtrIndicationType">
    <xs:attribute name="value" type="remoteCtrIndicationValueType" use="required"/>
</xs:complexType>
<!-- sipType -->
<xs:complexType name="sipType">
    <xs:choice>
        <xs:element name="active" type="booleanType"/>
        <xs:element name="alarm_enable" type="booleanType"/>
        <xs:element name="aliases_list" type="stringType"/>
        <xs:element name="auth" type="authType"/>
        <xs:element name="auth_qop" type="qopType"/>
        <xs:element name="digest" type="booleanType"/>
        <xs:element name="contact" type="stringType"/>
        <xs:element name="domain" type="stringType"/>
        <xs:element name="fork_mode" type="stringType"/>
        <xs:element name="fixed_contact" type="stringType"/>
        <xs:element name="group" type="stringType"/>
        <xs:element name="iface" type="stringType"/>
        <xs:element name="login" type="loginType"/>
        <xs:element name="nat_traversal" type="booleanType"/>
        <xs:element name="owner" type="stringType"/>
        <xs:element name="referred_by_as_cgpn" type="booleanType"/>
        <xs:element name="reg_expire_min" type="integerType"/>
        <xs:element name="reg_expire_max" type="integerType"/>
        <xs:element name="rfc4028_control" type="rfc4028ControlType"/>
        <xs:element name="routing_context" type="stringType"/>
        <xs:element name="modifier" type="stringType"/>
        <xs:element name="symbol_hash_as_is" type="booleanType"/>
        <xs:element name="trusted_ip" type="stringType">
            <xs:annotation>

```

```

        <xs:documentation xml:lang="en">
            Format:STRICT;IP:PORT;IP1-IP2:PORT_1-PORT2;IP3;...
            Sample:false;192.168.23.87:9988;192.168.23.181:9900-9910;192.168.23.0
        </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="tel_uri_in_diversion" type="booleanType"/>
<xs:element name="tel_uri_using" type="booleanType"/>
<xs:element name="sip_domain" type="stringType"/>
<xs:element name="sip_modifications" type="stringType"/>
<xs:element name="req100rel" type="booleanType"/>
<xs:element name="dtmf_relay" type="booleanType"/>
<xs:element name="original_cdpn_to" type="booleanType"/>
<xs:element name="my_from" type="stringType"/>
<xs:element name="transit" type="stringType"/>
<xs:element name="auto_answer_version" type="stringType"/>
<xs:element name="display_encoding" type="displayEncodingType"/>
<xs:element name="remote_ctr_indication" type="remoteCtrIndicationType"/>
<xs:element name="registered_domain_to_invite" type="booleanType"/>
<xs:element name="category_to_sip" type="stringType"/>

</xs:choice>
<xs:attribute name="domain" type="xs:string" use="required"/>
<xs:attribute name="group" type="xs:string" use="required"/>
<xs:attribute name="id" type="xs:string" use="optional"/>

</xs:complexType>
<!-- in -->
<xs:element name="in">
    <xs:complexType>
        <xs:all>
            <xs:element name="sip" type="sipType" minOccurs="0"/>
        </xs:all>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/sip_user_set

Пример 1:

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="sip_user_set.xsd">
  <sip group="auth_user" id="240901@biysk.local" domain="biysk.local">
    <login login="login_as_number" password="gt6_yrb45" />
  </sip>
</in>
```

Ответ: 201

Пример 2:

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="sip_user_set.xsd">
  <sip group="auth_user" id="240462@biysk.local" domain="biysk.local">
    <reg_expire_max value="850"/>
  </sip>
</in>
```

Ответ: 201

Пример 3:

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="sip_user_set.xsd">
  <sip group="auth_user" id="240462@biysk.local" domain="biysk.local">
    <routing_context value="ctx_from_ewsd"/>
  </sip>
</in>
```

Ответ: 201

Ис sip_user_remove

Команда позволяет удалить SIP-абонента.

Аналог команды в CoCon:

domain/<DOMAIN>/sip/user/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sip_user_remove

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 300 - в случае запроса подтверждения на удаления SIP-пользователя;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- nonEmptyString -->
  <xs:simpleType name="nonEmptyString">
    <xs:restriction base="xs:string">
      <xs:minLength value="1"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- aliasType-->
  <xs:complexType name="aliasType">
    <xs:attribute name="domain" type="nonEmptyString" use="required" />
    <xs:attribute name="iface" type="nonEmptyString" use="optional" />
    <xs:attribute name="group" type="nonEmptyString" use="required" />
    <xs:attribute name="force" type="xs:boolean" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Obsolete attribute. Use force attribute at request element
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>

  <!-- aliasesType-->
  <xs:complexType name="aliasesType">
    <xs:sequence>
      <xs:element name="alias" type="aliasType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="aliases" type="aliasesType"/>
            </xs:sequence>
            <xs:attribute name="force" type="xs:boolean" use="optional" default="true" />
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```



```

<xs:element name="progress">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="ref" minOccurs="1" maxOccurs="1" type="valueType" />
      <xs:element name="text" minOccurs="1" maxOccurs="1" type="valueType" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: `http://192.168.1.21:9999/commands/sip_user_remove`

Чтобы не отвечать на дополнительные вопросы, необходимо указать параметр `force="true"`.

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="sip_user_remove.xsd">
  <request>
    <aliases>
      <alias domain="biysk.local" group="loc.gr" iface="240600@biysk.local" force="true" />
      <alias domain="biysk.local" group="loc.gr" iface="240601@biysk.local" force="true" />
    </aliases>
  </request>
</in>

```

Ответ: 204

[Hc_sip_user_check](#)

Команда позволяет проверить доступность SIP-абонента(ов) посылкой SIP-запроса OPTIONS.

Аналог команды в CoCon:

`domain/<DOMAIN>/sip/user/check`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/sip_user_check`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- userType -->
  <xs:complexType name="userType">
    <xs:attribute name="accessible" type="xs:boolean" use="required"/>
    <xs:attribute name="iface" type="xs:string" use="required"/>
    <xs:attribute name="info" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- nonEmptyString -->
  <xs:simpleType name="nonEmptyString">
    <xs:restriction base="xs:string">
      <xs:minLength value="1"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- usersType -->
  <xs:complexType name="usersType">
    <xs:sequence>
      <xs:element name="user" type="userType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="user" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="group" type="nonEmptyString" use="required"/>
          <xs:attribute name="iface" type="nonEmptyString" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:all>
      <xs:element name="users" type="usersType" minOccurs="0" maxOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType" minOccurs="1" maxOccurs="1"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">

```

```

<xs:complexType>
  <xs:all>
    <xs:element name="response" type="responseType" minOccurs="0" maxOccurs="1"/>
  </xs:all>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/sip_user_check

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_sip_u
ser_check.xsd">
  <request domain="biysk.local">
    <user group="loc.gr" iface="240501@biysk.local"/>
    <user group="loc.gr" iface="240502@biysk.local"/>
    <user group="loc.gr" iface="240471@biysk.local"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sip_user_c
heck.xsd">
  <response>
    <users>
      <user iface="240501@biysk.local" accessible="true" info="OPTIONS to <sip:
240501@192.168.2.26>: 403 Forbidden, cseq: 202973 OPTIONS. Request by node: sip1@ecss1"/>
      <user iface="240502@biysk.local" accessible="true" info="OPTIONS to <sip:
240502@192.168.2.26>: 403 Forbidden, cseq: 526893 OPTIONS. Request by node: sip1@ecss1"/>
      <user iface="240471@biysk.local" accessible="false" info="User isn't registered now"/>
    >
    </users>
  </response>
</out>

```

Нс_snmp_agent_system_options - Управление настройками SNMP на уровне медиатора-а

- [Hc_snmp_agent_system_options_clean](#)
- [Hc_snmp_agent_system_options_info](#)
- [Hc_snmp_agent_system_options_set](#)

[Hc_snmp_agent_system_options_clean](#)

Команда сброса настроек подсистемы SNMP на уровне кластера медиатора.

Аналог команды в CoCon:

`cluster/mediator/<MD>/snmp/agent/properties/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/snmp_agent_system_options_clean`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean cluster options of snmp agent.
      Usage: http://server:9999/commands/snmp_agent_system_options_clean
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterCleanOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/snmp_agent_system_options_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="snmp_agent_system_options_clean.xsd">
  <request>
    <cluster_options>
      <cluster name="md1">
        <option name="snmp_v2c_community"/>
        <option name="snmp_v2c_enabled"/>
        <option name="snmp_v3_auth_priv"/>
        <option name="snmp_v3_enabled"/>
      </cluster>
    </cluster_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="snmp_agent_system_options_clean.xsd">
  <result>
    <cluster_options>
      <cluster name="md1">
        <success option="snmp_v3_enabled"/>
        <success option="snmp_v3_auth_priv"/>
        <success option="snmp_v2c_enabled"/>
        <success option="snmp_v2c_community"/>
      </cluster>
    </cluster_options>
  </result>
</out>

```

[Hc_snmp_agent_system_options_info](#)

Команда просмотра настроек подсистемы SNMP на уровне кластера медиатора.

Аналог команды в CoCon:

cluster/mediator/<MD>/snmp/agent/properties/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/snmp_agent_system_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get cluster options of snmp agent.
      Usage: http://server:9999/commands/snmp\_agent\_system\_options\_info
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <!-- snmpAgentClusterOptionsType -->
  <xs:complexType name="snmpAgentClusterOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="snmp_v2c_community" type="nonEmptyNormalizedStringType"/>
        <xs:attribute name="snmp_v2c_enabled" type="booleanType"/>
        <xs:attribute name="snmp_v3_auth_priv" type="nonEmptyNormalizedStringType"/>
        <xs:attribute name="snmp_v3_enabled" type="booleanType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterInfoInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterInfoOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>

```



```

    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/snmp_agent_system_options_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request>
    <cluster_options cluster="md1"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="snmp_agent_system_options_info.xsd">
  <result>
    <cluster_options>
      <cluster name="md1">
        <options xs:type="snmpAgentClusterOptionsType" snmp_v2c_community="none"
          snmp_v2c_enabled="false" snmp_v3_auth_priv="none" snmp_v3_enabled="false"/>
      </cluster>
    </cluster_options>
  </result>
</out>

```

[Hc_snmp_agent_system_options_set](#)

Команда изменения настроек подсистемы SNMP на уровне кластера медиатора.

Аналог команды в CoCon:

cluster/mediator/<MD>/snmp/agent/properties/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/snmp_agent_system_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set cluster options of snmp agent.
      Usage: http://server:9999/commands/snmp_agent_system_options_set
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <!-- snmpAgentClusterOptionsType -->
  <xs:complexType name="snmpAgentClusterOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="snmp_v2c_community" type="nonEmptyNormalizedStringType"/>
        <xs:attribute name="snmp_v2c_enabled" type="booleanType"/>
        <xs:attribute name="snmp_v3_auth_priv" type="nonEmptyNormalizedStringType"/>
        <xs:attribute name="snmp_v3_enabled" type="booleanType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterSetOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />

```

```

    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/snmp_agent_system__options_set

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="snmp_agent_system_options_set.xsd">
  <request>
    <cluster_options>
      <cluster name="md1">
        <options snmp_v2c_community="none" snmp_v2c_enabled="false"
snmp_v3_auth_priv="none" snmp_v3_enabled="false" xs:type="snmpAgentClusterOptionsType"/>
      </cluster>
    </cluster_options>
  </request>
</in>

```

Ответ:200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="snmp_agent_system_options_set.xsd">
  <result>
    <cluster_options>
      <cluster name="md1">
        <success option="snmp_v3_enabled"/>
        <success option="snmp_v3_auth_priv"/>
        <success option="snmp_v2c_enabled"/>
        <success option="snmp_v2c_community"/>
      </cluster>
    </cluster_options>
  </result>
</out>

```

Hc_sorm_options - Управление доменными настройками SNMP

- [Hc_sorm_options_clean](#)
- [Hc_sorm_options_info](#)
- [Hc_sorm_options_set](#)

Ис_sorm_options_clean

Команда сброса доменных настроек СОРМ.

Аналог команды в CoCon:

`domain/<DOMAIN>/sorm/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sorm_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean sorm options.
      Usage: http://server:9999/commands/sorm_options_clean.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/sorm_options_clean

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sorm_options_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="test.ats">
        <option name="adaptation"/>
        <option name="b_digits_mode"/>
        <option name="enabled"/>
        <option name="full_dial_number"/>
        <option name="password"/>
        <option name="platform_id"/>
        <option name="private_phone_prefix"/>
        <option name="radius_servers"/>
        <option name="send_optional_cf_notification"/>
        <option name="strict_intercity_numbers"/>
        <option name="strict_private_numbers"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sorm_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="adaptation"/>
        <success option="b_digits_mode"/>
        <success option="enabled"/>
        <success option="full_dial_number"/>
        <success option="password"/>
        <success option="platform_id"/>
        <success option="private_phone_prefix"/>
        <success option="radius_servers"/>
        <success option="send_optional_cf_notification"/>
        <success option="strict_intercity_numbers"/>
        <success option="strict_private_numbers"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_sorm_options_info

Команда просмотра информации о доменных настройках СОРМ.

Аналог команды в CoCon:

domain/<DOMAIN>/sorm/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sorm_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show sorm options.
      Usage: http://server:9999/commands/sorm\_options\_info.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- prikazType -->
  <xs:simpleType name="prikazType">
    <xs:restriction base="unionType">
      <xs:enumeration value="70"/>
      <xs:enumeration value="268"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- sormOptionsType -->
  <xs:complexType name="sormOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="adaptation" type="atomType"/>
        <xs:attribute name="b_digits_mode" type="atomType"/>
        <xs:attribute name="enabled" type="booleanType"/>
        <xs:attribute name="full_dial_number" type="booleanType"/>
        <xs:attribute name="password" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              In case of strong security restrictions password may be shown as
              *****
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="platform_id" type="stringType"/>
        <xs:attribute name="private_phone_prefix" type="stringType"/>
        <xs:attribute name="radius_servers" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Radius server option:
              Format: IP:Secret;IP2:Secret2
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="send_optional_cf_notification" type="booleanType"/>
        <xs:attribute name="strict_intercity_numbers" type="booleanType"/>
        <xs:attribute name="strict_private_numbers" type="booleanType"/>
        <xs:attribute name="prikaz" type="prikazType"/>
        <xs:attribute name="city_prefix_70" type="stringType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

        </xs:extension>
    </xs:complexContent>
</xs:complexType>

<!-- in -->
<xs:element name="in">
    <xs:complexType>
        <xs:all>
            <xs:element name="request">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainInfoInType"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
        </xs:all>
    </xs:complexType>
</xs:element>
<!-- in -->

<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="result">
                <xs:complexType>
                    <xs:all>
                        <xs:element name="domain_options" type="domainInfoOutType"/>
                    </xs:all>
                </xs:complexType>
            </xs:element>
            <xs:element name="error" type="errorType" />
        </xs:choice>
    </xs:complexType>
</xs:element>
<!-- out -->
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sorm_options_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="sorm_options_info.xsd">
  <request>
    <domain_options domain="ct.office"/>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sorm_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="sormOptionsType"/>
      </system>
      <domain name="ct.office">
        <options xs:type="sormOptionsType" strict_intercity_numbers="false"
          send_optional_cf_notification="true" radius_servers="192.168.1.12:secret;10.16.33.5:secret2"
          private_phone_prefix="0" password="pasS1234" enabled="true" b_digits_mode="routing"
          adaptation="default_adaptation"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Нс_sorm_options_set

Команда изменения доменных настроек СОПМ.

Аналог команды в CoCon:

domain/**<DOMAIN>**/**sorm/set**

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sorm_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show sorm options.
      Usage: http://server:9999/commands/sorm_options_info.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- prikazType -->
  <xs:simpleType name="prikazType">
    <xs:restriction base="unionType">
      <xs:enumeration value="70"/>
      <xs:enumeration value="268"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- sormOptionsType -->
  <xs:complexType name="sormOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="adaptation" type="atomType"/>
        <xs:attribute name="b_digits_mode" type="atomType"/>
        <xs:attribute name="enabled" type="booleanType"/>
        <xs:attribute name="full_dial_number" type="booleanType"/>
        <xs:attribute name="password" type="stringType"/>
        <xs:attribute name="platform_id" type="stringType"/>
        <xs:attribute name="private_phone_prefix" type="stringType"/>
        <xs:attribute name="radius_servers" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Radius server option:
              Format: IP:Secret;IP2:Secret2
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="send_optional_cf_notification" type="booleanType"/>
        <xs:attribute name="strict_intercity_numbers" type="booleanType"/>
        <xs:attribute name="strict_private_numbers" type="booleanType"/>
        <xs:attribute name="prikaz" type="prikazType"/>
        <xs:attribute name="city_prefix_70" type="stringType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>

```

```

    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- in -->

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetOutType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
      <xs:element name="error" type="errorType" />
    </xs:choice>
  </xs:complexType>
</xs:element>
<!-- out -->
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/sorm_options_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sorm_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="sormOptionsType"/>
      </system>
      <domain name="ct.office">
        <options xs:type="sormOptionsType" strict_intercity_numbers="false"
        send_optional_cf_notification="true" radius_servers="192.168.1.12:secret;10.16.33.5:secret2"
        private_phone_prefix="0" password="pasS1234" enabled="true" b_digits_mode="routing"
        adaptation="default_adaptation"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sorm_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="ct.office">
        <success option="adaptation"/>
        <success option="b_digits_mode"/>
        <success option="enabled"/>
        <success option="password"/>
        <success option="private_phone_prefix"/>
        <success option="radius_servers"/>
        <success option="send_optional_cf_notification"/>
        <success option="strict_intercity_numbers"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_sorm_system_options - Управление системными свойствами СОРМ

- [Hc_sorm_system_options_clean](#)
- [Hc_sorm_system_options_info](#)
- [Hc_sorm_system_options_set](#)

[Hc_sorm_system_options_clean](#)

Команда сброса системных настроек COPM.

Аналог команды в CoCon:

`sorm/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sorm_system_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean system sorm options.
      Usage: http://server:9999/commands/sorm_system_options_clean.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- sormSystemOptionsType -->
  <xs:complexType name="sormSystemOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="aggregate_sorm_settings" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              usage:
                Name:Password:Domain1,Domain2;Name2:Password2:Domain12,Domain22
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="ssh_dir" type="stringType"/>
        <xs:attribute name="ssh_listen_interface" type="stringType"/>
        <xs:attribute name="ssh_port" type="inetPortType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>

```

```

        <xs:element name="result">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemCleanOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/sorm_system_options_clean

```

<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sorm_system_options_clean.xsd">
  <request>
    <system_options>
      <system>
        <option name="aggregate_sorm_settings"/>
        <option name="ssh_dir"/>
        <option name="ssh_listen_interface"/>
        <option name="ssh_port"/>
      </system>
    </system_options>
  </request>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sorm_system_options_clean.xsd">
  <result>
    <system_options>
      <system>
        <success option="ssh_port"/>
        <success option="ssh_listen_interface"/>
        <success option="ssh_dir"/>
        <success option="aggregate_sorm_settings"/>
      </system>
    </system_options>
  </result>
</out>
```

Нс_sorm_system_options_info

Команда просмотра информации о системных настройках СОРМ.

Аналог команды в CoCon:

sorm/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sorm_system_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show system sorm options.
      Usage: http://server:9999/commands/sorm\_system\_options\_info.
    </xs:documentation>
  </xs:annotation>
</xs:annotation>
<xs:include schemaLocation="error.xsd"/>
<xs:include schemaLocation="ecm_common.xsd"/>

<!-- sormSystemOptionsType -->
<xs:complexType name="sormSystemOptionsType">
  <xs:complexContent>
    <xs:extension base="baseOptionsType">
      <xs:attribute name="aggregate_sorm_settings" type="stringType">
        <xs:annotation>
          <xs:documentation xml:lang="en">
            usage:
              Name:Password:Domain1,Domain2;Name2:Password2:Domain12,Domain22
              In case of strong security restrictions password may be shown as
            *****
          </xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="ssh_dir" type="stringType"/>
      <xs:attribute name="ssh_listen_interface" type="stringType"/>
      <xs:attribute name="ssh_port" type="inetPortType"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="system_options" type="systemInfoInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>

<!-- out -->
<xs:element name="out">

```

```

<xs:complexType>
  <xs:choice>
    <xs:element name="result">
      <xs:complexType>
        <xs:all>
          <xs:element name="system_options" type="systemInfoOutType"/>
        </xs:all>
      </xs:complexType>
    </xs:element>
    <xs:element name="error" type="errorType"/>
  </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/sorm_system_options_info

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="sorm_system_options_info.xsd">
  <request>
    <system_options/>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="sorm_system_options_info.xsd">
  <result>
    <system_options>
      <system>
        <options xs:type="sormSystemOptionsType"
  aggregate_sorm_settings="user:*****:ct.office" ssh_dir="/etc/ecss/ssh"
  ssh_listen_interface="0.0.0.0" ssh_port="7777"/>
      </system>
    </system_options>
  </result>
</out>

```

Ис_sorm_system_options_set

Команда изменения системных настроек СОРМ.

Аналог команды в CoCon:

sorm/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sorm_system_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set system sorm options.
      Usage: http://server:9999/commands/sorm_system_options_set.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- sormSystemOptionsType -->
  <xs:complexType name="sormSystemOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="aggregate_sorm_settings" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              usage:
                Name:Password:Domain1,Domain2;Name2:Password2:Domain12,Domain22
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="ssh_dir" type="stringType"/>
        <xs:attribute name="ssh_listen_interface" type="stringType"/>
        <xs:attribute name="ssh_port" type="inetPortType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>

```

```

        <xs:element name="result">
            <xs:complexType>
                <xs:all>
                    <xs:element name="system_options" type="systemSetOutType"/>
                </xs:all>
            </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" />
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/sorm_system_options_set

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sorm_system_options_set.xsd">
  <request>
    <system_options>
      <system>
        <options xs:type="sormSystemOptionsType"
  aggregate_sorm_settings="user:password:test.ats;user2:password:biysk.local, domain.second"
  ssh_dir="/etc/ecss/ssh" ssh_listen_interface="::1" ssh_port="7777"/>
      </system>
    </system_options>
  </request>
</in>

```

Ответ: 200


```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sorm_system_options_set.xsd">
  <result>
    <system_options>
      <system>
        <success option="ssh_port"/>
        <success option="ssh_listen_interface"/>
        <success option="ssh_dir"/>
        <success option="aggregate_sorm_settings"/>
      </system>
    </system_options>
  </result>
</out>

```

Нс_sp_api - Команды httpTerminal-а для работы с сессией пользователя в Портале абонента

- [Нс_sp_api_login](#)
- [Нс_sp_api_logout](#)
- [Нс_sp_api_change_login](#)
- [Нс_sp_api_change_password](#)
- [Нс_sp_api_check_login](#)
- [Нс_sp_api_check_session](#)

Нс_sp_api_login

Команда регистрации пользователя в портале абонента по домену, логину, паролю.

Аналога команды в CoCon нет.

Метод http запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/sp_login

Коды ответа:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="login" type="xs:string" use="required"/>
    <xs:attribute name="password" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok" type="okType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>

  <xs:complexType name="okType">
    <xs:attribute name="session" type="xs:string" use="required"/>
    <xs:attribute name="alias_address" type="xs:string"/>
    <xs:attribute name="alias_iface" type="xs:string"/>
  </xs:complexType>

  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:Запрос: http://192.168.1.21:9999/commands/sp_login

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="sp_api_login.xsd">
  <request domain="biysk.local" login="240101" password="ssw_101" />
</in>
```

Ответ:200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sp_api_login.xsd">
  <response>
    <ok session="b642dd1dd8500124fc4593ebfb3636d1daddbdfa" alias_iface="062743c4240eae38"
    alias_address="240101"/>
  </response>
</out>
```

Если пользователя с указанной парой логин/пароль не существует:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sp_api_login.xsd">
  <response>
    <error description="invalid pair login/password"/>
  </response>
</out>
```

Нс_sp_api_logout

Команда позволяет по указанному идентификатору завершить сессию с "Порталом абонента" в определенном домене.

Аналога команды в CoCon нет.**Метод http запроса:**

POST

Шаблон HTTP-запроса:http://server:9999/commands/sp_logout

Коды ответа:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="session" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok" type="okType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>

  <xs:complexType name="okType" />

  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:Запрос: http://192.168.1.21:9999/commands/sp_logout

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="sp_api_1
  ogout.xsd">
  <request domain="biysk.local" session="b642dd1dd8500124fc4593ebfb3636d1daddbdfa" />
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sp_api_log
  out.xsd">
  <response>
    <ok/>
  </response>
</out>
```

Ис_sp_api_change_login

Команда позволяет по указанному логину пользователя "Портала абонента" сменить его на новый в определенном домене

Аналог команды в CoCon:**/domain/<DOMAIN>/alias/set****Метод http запроса:**

POST

Шаблон HTTP-запроса:http://server:9999/commands/sp_change_login**Коды ответа:**

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="oldlogin" type="xs:string" use="required"/>
    <xs:attribute name="newlogin" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok" type="okType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>

  <xs:complexType name="okType" />

  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sp_change_login

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="sp_api_change_login.xsd">
  <request domain="biysk.local" oldlogin="240101" newlogin="240101_ssw" />
</in>
```

Ответ: 200

Успешная замена логина:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sp_api_change_login.xsd">
  <response>
    <ok/>
  </response>
</out>
```

Если пользователя с указанным логином не существует:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sp_api_change_login.xsd">
  <response>
    <error description="not_exists"/>
  </response>
</out>
```

Ис_sp_api_change_password

Команда позволяет по указанному логину пользователя "Портала абонента" сменить пароль на новый в определенном домене

Аналог команды в CoCon:

/domain/<DOMAIN>/alias/set

Шаблон HTTP-запроса:

http://server:9999/commands/sp_change_password

Коды ответа:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="login" type="xs:string" use="required"/>
    <xs:attribute name="newpassword" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok" type="okType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>

  <xs:complexType name="okType" />

  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/sp_change_password

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="sp_api_change_password.xsd">
  <request domain="biysk.local" login="240466_123" newpassword="456" />
</in>
```

Ответ: 200

Успешная смена пароля:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sp_api_change_password.xsd">
  <response>
    <ok/>
  </response>
</out>
```

Если пользователя с указанным логином не существует:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sp_api_change_password.xsd">
  <response>
    <error description="not_exists"/>
  </response>
</out>
```

Нс_sp_api_check_login

Команда позволяет проверить свободен ли указанный логин для "Портала абонента" в определенном домене

Аналога команды в CoCon нет.

Шаблон HTTP-запроса:

http://server:9999/commands/sp_check_login

Коды ответа:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="login" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok" type="okType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>

  <xs:complexType name="okType" />

  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/sp_check_login

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="sp_api_heck_login.xsd">
  <request domain="biysk.local" login="240466" />
</in>
```

Ответ: 200

Если не существует пользователя с указанным логином.

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sp_api_heck_login.xsd">
  <response>
    <ok/>
  </response>
</out>
```

Если существует пользователь с указанным логином.

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sp_api_heck_login.xsd">
  <response>
    <error description="already_exists"/>
  </response>
</out>
```

Ис_sp_api_check_session

Команда позволяет проверить по указанному идентификатору и логину существует ли сессия с "Порталом абонента" в определенном домене

Аналога команды в CoCon нет.

Шаблон HTTP-запроса:

http://server:9999/commands/sp_check_session

Коды ответа:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="login" type="xs:string" use="required"/>
    <xs:attribute name="session" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:choice>
      <xs:element name="ok" type="okType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>

  <xs:complexType name="okType" />

  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/sp_check_session

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="sp_api_heck_session.xsd">
  <request domain="biysk.local" login="Login" session="7e67399706f6b734ec7806cacca9ee9ffe2e2845" />
</in>
```

Ответ: 200

Если сессия существует:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sp_api_heck_session.xsd">
  <response>
    <ok/>
  </response>
</out>
```

Если сессия не существует:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sp_api_heck_session.xsd">
  <response>
    <error description="not_exists"/>
  </response>
</out>
```

Нс_sp - Управление порталом (кабинетом) абонента

- Нс_check_connection
- Нс_sp_properties_list
- Нс_sp_properties_set
- Нс_sp_properties_clean
- Нс_sync

Нс_check_connection

Команда проверки доступности БД портала абонента.

Аналог команды в CoCon:

domain/<DOMAIN>/subscriber-portal/check-connection

Шаблон HTTP-запроса:

http://server:9999/commands/check_connection

Коды ответа:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="hc_check_connection">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Command for checking connection to subscriber portal database.
      Copyright (C) 2014, Eltex. All right reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="storage" type="nonEmptyTokenType" use="optional">
              <xs:annotation>
                <xs:documentation xml:lang="en">
                  Deprecated attribute.
                </xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="domain" type="nonEmptyTokenType" use="optional" />
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="success" minOccurs="1" maxOccurs="1" />
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- out -->
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/check_connection

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_check_connection.xsd">
  <request domain="biysk.local" />
</in>
```

Ответ: 200

Соединение с базой успешно установлено:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_check_connection.xsd">
  <success/>
</out>
```

Ответ: 404.

При соединении произошла ошибка (не заданы параметры портала абонента).

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_check_connection.xsd">
  <error cmd="check-connection" reason="db_error" body="conn_params_is_null"/>
</out>
```

hc_sp_properties_list

Команда просмотра свойств портала абонентов для всех виртуальных АТС.

Аналог команды в CoCon:

domain/<DOMAIN>/subscriber-portal/properties/info

Шаблон URL-запроса:

http://server:9999/commands/sp_properties_list

Коды ответа:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="sp_properties_list">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get list of subscriber portal properties for all active domains in cluster.
      Usage: http://server:9999/commands/sp_properties_list.
      Implementation: hc_sp_properties_list.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

<!-- Types -->

<!-- request -->
<xs:complexType name="requestType">
  <xs:attribute name="storage" type="nonEmptyTokenType" use="optional">
    <xs:annotation>
      <xs:documentation xml:lang="en">
        Deprecated attribute.
      </xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="domain" type="nonEmptyTokenType" use="optional" />
</xs:complexType>
<!-- request -->

<!-- successful response -->
<!-- spPropertiesType-->
<xs:complexType name="spDomainPropertiesType">
  <xs:complexContent>
    <xs:extension base="baseOptionsType">
      <xs:attribute name="web_sp_server" type="xs:string" use="optional" />
      <xs:attribute name="auto_create_account" type="xs:boolean" use="optional" />
      <xs:attribute name="auto_delete_account" type="xs:boolean" use="optional" />
      <xs:attribute name="connection" type="xs:normalizedString" use="optional" />
      <xs:attribute name="ldap_authentication_enabled" type="xs:boolean" use="optional" />
    </xs:extension>

    <xs:attribute name="ldap_server_id" type="xs:normalizedString" use="optional" >
      <xs:annotation>
        <xs:documentation xml:lang="en">
          LDAP/AD server id (from domain LDAP/AD servers' list
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:extension>

```

```

        </xs:complexContent>
    </xs:complexType>
    <!-- successful response -->

    <!-- resultType-->
    <xs:complexType name="resultType">
        <xs:all>
            <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
        </xs:all>
    </xs:complexType>

<!-- Root elements -->

    <!-- in -->
    <xs:element name="in">
        <xs:complexType>
            <xs:all>
                <xs:element name="request" type="requestType" minOccurs="1" maxOccurs="1" />
            </xs:all>
        </xs:complexType>
    </xs:element>
    <!-- in -->

    <!-- out -->
    <xs:element name="out">
        <xs:complexType>
            <xs:choice>
                <xs:element name="result" type="resultType"/>
                <xs:element name="error" type="errorType"/>
            </xs:choice>
        </xs:complexType>
    </xs:element>
    <!-- out -->

</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/sp_properties_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sp_properties_list.xsd">
    <request storage="ds1"/>
</in>

```

Ответ:

Успешный запрос.

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sp_properties_list.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="spDomainPropertiesType" web_sp_server="http://sp.web.ecss/"
  ldap_server_id="" ldap_authentication_enabled="false" connection="" auto_delete_account="false"
  auto_create_account="false"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="spDomainPropertiesType" connection="Host: 192.168.1.71; Port:
  3306; Login: subscribers; Password: subscribers; Database: ecss_subscribers"
  auto_delete_account="true" auto_create_account="true"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Ошибка при выполнении запроса:

```

<?xml version="1.0" encoding="UTF-8"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sp_properties_list.xsd">
  <error cmd="sp_properties_list" reason="error" body="test" />
</out>

```

[Hc_sp_properties_set](#)

Команда для редактирования свойств портала абонентов.

Аналог команды в CoCon:

domain/<DOMAIN>/subscriber-portal/properties/set

Шаблон URL-запроса:

http://server:9999/commands/sp_properties_set

Коды ответа:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="sp_properties_list">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set properties of subscriber portal.
      Usage: http://server:9999/commands/sp_properties_set.
      Implementation: hc_sp_properties_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="set" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="property" type="nonEmptyTokenType"
use="required" >
                    <xs:annotation>
                      <xs:documentation xml:lang="en">
                        Property connection:
                        Usage: Host: host; Port: port; Login: login;
                        Password: password; Database: database
                        host - The host of the MySQL server;
                        port - The port of the MySQL server;
                        login - MySQL server login;
                        password - MySQL server password;
                        database - MySQL server database
                        Sample: Host: localhost; Port: 3306; Login:
                        mysql; Password: passwd; Database: sp-db
                        Property web_sp_server: string
                        Property auto_create_account: boolean
                        Property auto_delete_account: boolean
                        Property ldap_server_id: string
                        Property ldap_authentication_enabled: boolean
                        Property enable: boolean
                      </xs:documentation>
                    </xs:annotation>
                  </xs:attribute>
                  <xs:attribute name="value" type="xs:normalizedString"
use="required" />

```

```

        </xs:complexType>
    </xs:element>
</xs:sequence>
<xs:attribute name="storage" type="nonEmptyTokenType" use="optional">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Deprecated attribute.
        </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="domain" type="nonEmptyTokenType" use="optional" />
</xs:complexType>
</xs:element>
</xs:all>
</xs:complexType>
</xs:element>
<!-- in -->

<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:sequence minOccurs="0" maxOccurs="unbounded">
            <xs:element name="success" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                    <xs:attribute name="property" type="xs:token" use="required" />
                    <xs:attribute name="old_value" type="xs:normalizedString" use="required"
/>

                    <xs:attribute name="new_value" type="xs:normalizedString" use="required"
/>

                </xs:complexType>
            </xs:element>
            <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded" />
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/sp_properties_set

Изменение всех свойств домена biysk.local:

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sp_properties_set.xsd">
  <request storage="ds1" domain="biysk.local">
    <set property="auto_create_account" value="true" />
    <set property="auto_delete_account" value="true" />
    <set property="connection" value="Host: localhost; Port: 3306; Login: user; Password:
secret; Database: test" />
  </request>
</in>
```

Ответ:200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sp_properties_set.xsd">
  <success property="auto_create_account" old_value="true" new_value="true"/>
  <success property="auto_delete_account" old_value="true" new_value="true"/>
  <success property="connection" old_value="{\"sp.mysql.ecss\",3306,\"subscribers\",\"subscribers\",\"e
css_subscribers\"}" new_value="{\"localhost\",3306,\"user\",\"secret\",\"test\"}"/>
</out>
```

Ис_sp_properties_clean

Команда для сброса политики работы с порталом абонентов.

Аналог команды в CoCon:

domain/<DOMAIN>**/subscriber-portal/properties/clean**

Шаблон URL-запроса:

http://server:9999/commands/sp_properties_clean

Коды ответа:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="sp_properties_clean">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Reset to default properties of subscriber portal.
      Usage: http://server:9999/commands/sp\_properties\_clean.
      Implementation: hc_sp_properties_clean.erl.
      Git: git@git.eltex.loc:8080/httpTerminal.git.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

<!-- Root elements -->

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:sequence minOccurs="0" maxOccurs="unbounded">
            <xs:element name="clean" minOccurs="0" maxOccurs="unbounded">
              <xs:complexType>
                <xs:attribute name="property" type="nonEmptyTokenType"
use="required"/>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
          <xs:attribute name="storage" type="nonEmptyTokenType" use="optional">
            <xs:annotation>
              <xs:documentation xml:lang="en">
                Deprecated attribute.
              </xs:documentation>
            </xs:annotation>
          </xs:attribute>
          <xs:attribute name="domain" type="nonEmptyTokenType" use="optional" />
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- in -->

<!-- out -->
<xs:element name="out">
  <xs:complexType>

```

```

        <xs:sequence minOccurs="0" maxOccurs="unbounded">
            <xs:element name="success" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                    <xs:attribute name="property" type="nonEmptyTokenType" use="required" />
                </xs:complexType>
            </xs:element>
            <xs:element name="error" type="errorType" minOccurs="0" maxOccurs="unbounded" />
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->

</xs:schema>

```

Пример:

Запрос на сброс всех свойств:

http://192.168.1.21:9999/commands/sp_properties_clean

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sp_properties_clean.xsd">
    <!-- Сбросить все свойства для всех доменов -->
    <request storage="ds1" />
</in>

```

Сброс всех свойств, второй вариант (если указать домен, будут сброшены все свойства для этого домена):

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sp_properties_clean.xsd">
    <!-- Сбросить все свойства для всех доменов, второй вариант -->
    <request storage="ds1" domain="biysk.local" />
</in>

```

Ответ на запрос о сбросе всех свойств:

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="sp_properties_clean.xsd">
    <success property="all"/>
</out>

```

Запрос на сброс некоторых свойств для домена:


```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sp_properties_clean.xsd">
  <!-- Сбросить перечисленные свойства для d.local -->
  <request storage="ds1" domain="biysk.local">
    <clean property="auto_create_account" />
    <clean property="auto_delete_account" />
  </request>
</in>
```

Ответ на запрос о сбросе некоторых свойств для определенного домена:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="sp_properties_clean.xsd">
  <success property="auto_create_account"/>
  <success property="auto_delete_account"/>
</out>
```

Ответ на запрос о сбросе трех свойств, при обработке одного из которых произошла ошибка:

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="sp_properties_clean.xsd">
  <success property="auto_create_account"/>
  <success property="auto_delete_account"/>
  <error cmd="sp_properties_clean" reason="error" body="{error,{}" entity="connection"/>
</out>
```

Ис_sync

Синхронизация алиасов ECSS-10 с порталом абонентов.

Аналог команды в CoCon:

domain/<DOMAIN>/subscriber-portal/sync

Шаблон URL-запроса:

http://server:9999/commands/sync

Коды ответа:

- 200 - в случае успеха;
- 404 - в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="sync">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Synchronizing aliases of local softswitch and aliases of subscriber portal.
      Usage: http://server:9999/commands/sync.
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <!-- resultType -->
  <xs:simpleType name="resultType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="ok" />
      <xs:enumeration value="already_exists" />
      <xs:enumeration value="error" />
    </xs:restriction>
  </xs:simpleType>
  <!-- /resultType -->

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="nonEmptyTokenType" use="required" />
            <xs:attribute name="address_range" type="nonEmptyTokenType"
use="required" />
            <xs:attribute name="iface_range" type="nonEmptyTokenType" use="required"
/>
            <xs:attribute name="password" type="nonEmptyTokenType" use="optional" />
            <xs:attribute name="verbose" type="xs:boolean" use="optional" />
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- /in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="results" minOccurs="1" maxOccurs="1">
          <xs:complexType>

```

```

        <xs:sequence>
          <xs:element name="result" minOccurs="0" maxOccurs="unbounded">
            <xs:complexType>
              <xs:attribute name="alias" type="xs:string" use="required" />
              <xs:attribute name="login" type="xs:string" use="required" />
              <xs:attribute name="password" type="xs:string" use="required" />
            </xs:complexType>
            <xs:attribute name="result" type="resultType" use="required" />
          </xs:element>
        </xs:sequence>
        <xs:attribute name="success" type="xs:nonNegativeInteger" use="required" />
        <xs:attribute name="fail" type="xs:nonNegativeInteger" use="required" />
        <xs:attribute name="total" type="xs:nonNegativeInteger" use="required" />
      </xs:complexType>
    </xs:element>
    <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
  </xs:choice>
</xs:complexType>
</xs:element>
<!-- /out -->
</xs:schema>

```

Запрос:

<http://server:9999/commands/sync>

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sync.xsd">
  <request domain="biysk.local"
    address_range="24047{3-4}"
    iface_range="24047{3-4}@biysk.local"
    password="password"
    verbose="true" />
</in>

```

Ответ:200

Успешный запрос:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="sync.xsd">
  <results success="2" fail="0" total="2">
    <result alias="240474, 06201d996900719e" login="240474" password="password" result="ok"/>
    <result alias="240473, 06201d99d02a910b" login="240473" password="password" result="ok"/>
  </results>
</out>
```

Ошибка при выполнении запроса:

```
<?xml version="1.0" encoding="UTF-8"?><out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:noNamespaceSchemaLocation="sync.xsd">
  <error cmd="sync" reason="no_enough_addresses" body="{ }"/>
</out>
```

Нс_ss_alias_package_list - Команда для просмотра пакетов ДВО абонента

Нс_ss_package_list

Команда для просмотра информации об установленных пакетах услуг у абонента(ов)

Аналог команды в CoСon:

```
domain/<DOMAIN>/ss/limit --show-aliases
cluster/storage/<DS_CLUSTER>/ss/limit --show-aliases
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_package_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки (с указанием ошибки в теле ответа).

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="packageInfoType">
    <xs:attribute name="package_name" type="xs:string"/>
    <xs:attribute name="allocate_time" type="xs:integer"/>
    <xs:attribute name="is_active" type="xs:boolean"/>
    <xs:attribute name="is_priority" type="xs:boolean"/>
    <xs:attribute name="description" type="xs:string"/>
    <xs:attribute name="ss_list" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="warningType">
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="aliasInfoType">
    <xs:choice>
      <xs:element name="package_info" type="packageInfoType" maxOccurs="unbounded"
minOccurs="0"/>
      <xs:element name="warning" type="warningType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:choice>
    <xs:attribute name="address" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="resultType">
    <xs:sequence maxOccurs="unbounded" minOccurs="0">
      <xs:element name="alias_info" type="aliasInfoType" maxOccurs="unbounded" minOccurs="0"
"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string"/>
            <xs:attribute name="addresses" type="xs:string"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>

```

```

        <xs:choice>
            <xs:element name="result" type="resultType"/>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Примеры

Запрос:

http://192.168.1.21:9999/commands/ss_package_list

```

<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss_alias_package_list.xsd">
    <request domain="biysk.local" addresses="{240101,240464}"/>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss_alias_package_list.xsd">
    <result>
        <alias_info address="240101">
            <package_info package_name="ECSS-ADV" is_active="true" is_priority="false"
                allocate_time="1588648026947207783" description="Дополнительные услуги"
            ss_list=
            "blf,call_recording,chunt,clir,cliro,conference,fax_to_email,mcid,mgm,presence,ring_back_tone,voi
            cemail"/>
        </alias_info>
        <alias_info address="240464">
            <package_info package_name="ECSS-ADV" is_active="true" is_priority="false"
                allocate_time="1588647997838303176" description="Дополнительные услуги"
            ss_list=
            "blf,call_recording,chunt,clir,cliro,conference,fax_to_email,mcid,mgm,presence,ring_back_tone,voi
            cemail"/>
            <package_info package_name="ECSS-BAS+" is_active="true" is_priority="false"
                allocate_time="1588652854940887353" description="Расширение базовых услуг"
            ss_list=
            "acb,alarm,autoresdial,callback,cgg,ctr,cw,direct_call,dnd,intervention,personal_ivr,rbp,rfc,sca,s
            co_black,sco_white,scr,voice_page,zone_page"/>
        </alias_info>
    </result>
</out>

```

Hc_ss_domain_profile - Управление профилями услуг

- [Hc_ss_domain_profile_activate](#)
- [Hs_ss_domain_profile_add_ss](#)
- [Hc_ss_domain_profile_copy](#)
- [Hc_ss_domain_profile_info](#)
- [Hc_ss_domain_profile_list](#)
- [Hc_ss_domain_profile_remove](#)
- [Hc_ss_domain_profile_set](#)

[Hc_ss_domain_profile_activate](#)

Команда позволяет активировать услуги доменного профиля на заданном адресе.

Аналог команды в CoCon:

`/domain/<DOMAIN>/ss/profile/activate`

Шаблон http запроса:

`http://host:port/commands/ss_domain_profile_activate`

Код http ответа:

200 – в случае успеха.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <!-- activatePropertyType -->

  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- activateType -->

  <xs:complexType name="activateType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="profile" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- okType -->

  <xs:complexType name="okType">
    <xs:attribute name="address" type="xs:string" use="optional"/>
    <xs:attribute name="profile" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- errorType -->

  <xs:complexType name="errorType">
    <xs:attribute name="address" type="xs:string" use="optional"/>
    <xs:attribute name="profile" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- requestType -->

  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="activate" type="activateType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="force" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- in -->

```



```

<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request" type="requestType" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- out -->

<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="ok" type="okType" minOccurs="0"
maxOccurs="unbounded"/>
            <xs:element name="error" type="errorType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Код http ответа:

200 – в случае успеха;
404 – в случае ошибки.

Пример 1**Запрос:**

http://192.168.1.21:9999/commands/ss_domain_profile_activate

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_ss_domain_profile_activate.xsd">
  <request domain="biysk.local">
    <activate profile="work_ss" address="240473"></activate>
    <activate profile="work_ss" address="240469"></activate>
  </request>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ss_domain_profile_activate.xsd">
  <response>
    <ok address="240469" profile="work_ss"/>
    <ok address="240473" profile="work_ss"/>
  </response>
</out>
```

[Hs_ss_domain_profile_add_ss](#)

Команда позволяет добавить новый набор услуг в уже созданные доменные профили.

Аналог команды в CoCon:

/domain/<DOMAIN>/ss/profile/ss-add

Шаблон http запроса:

http://host:port/commands/ss_domain_profile_add_ss

Код http ответа:

201 – в случае успеха;

200 – в случае ошибки (с указанием ошибки в теле ответа).

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- propertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          SS property's value, formatted by in the ss_activate command.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- ssType -->
  <xs:complexType name="ssType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="enable" type="xs:boolean" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
  </xs:complexType>
  <!-- setProfileType -->
  <xs:complexType name="setProfileType">
    <xs:sequence>
      <xs:element name="ss" type="ssType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Domain's profile name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="profile" type="setProfileType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Domain name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>

```

```

        </xs:attribute>
    </xs:complexType>
    <!-- in -->
    <xs:element name="in">
        <xs:complexType>
            <xs:all>
                <xs:element name="request" type="requestType"/>
            </xs:all>
        </xs:complexType>
    </xs:element>
    <!-- out -->
    <xs:element name="out">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/ss_domain_profile_add_ss

```

<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_ss_domain_profile_add_ss.xsd">
  <request domain="biysk.local">
    <profile name="ss_test">
      <ss name="ctr" enable="true" active="true"></ss>
    </profile>
  </request>
</in>

```

Ответ: 201

[Hc_ss_domain_profile_copy](#)

Команда позволяет сделать копию доменного профиля с новым именем.

Аналог команды в CoCon:

`/domain/<DOMAIN>/ss/profile/copy`

Шаблон http запроса:

`http://host:port/commands/ss_domain_profile_copy`

Код http ответа:

201 – в случае успеха;

200 – в случае ошибки (с указанием ошибки в теле ответа).

Xsd схема XML запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:include schemaLocation="error.xsd"/>

  <!-- requestType -->

  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="src_name" type="xs:string" use="required"/>
    <xs:attribute name="dst_name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->

  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

</xs:schema>
```

Пример 1

Запрос:

http://192.168.1.21:9999/commands/ss_domain_profile_copy

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_ss_domain_profile_copy.xsd">
  <request domain="biysk.local" src_name="min_ss" dst_name="min_ss_copy"/>
</in>
```

Ответ:

201

Пример 2

В случае, если профиль с именем min_ss_copy уже существует

Ответ:

200

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ss_domain_profile_copy.xsd">
  <error cmd="hc_ss_domain_profile_copy" reason="already_exists" body="{error,already_exists}"/>
</out>
```

Пример 3

В случае, если исходный профиль не существует

Запрос: http://192.168.1.21:9999/commands/ss_domain_profile_copy

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_ss_domain_profile_copy.xsd">
  <request domain="biysk.local" src_name="unknown_profile" dst_name="to"/>
</in>
```

Ответ:

200

```
?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ss_domain_profile_copy.xsd">
  <error cmd="hc_ss_domain_profile_copy" reason="not_exists" body="{error,not_exists}"/>
</out>
```

Нс_ss_domain_profile_info

Команда позволяет получить настройки доменного профиля услуги по имени.

Аналог команды в CoCon:

/domain/<DOMAIN>/ss/profile/info

Шаблон http запроса:

http://host:port/commands/ss_domain_profile_info

Код http ответа:

200 – в случае успеха.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="ss.xsd"/>
  <xs:include schemaLocation="error.xsd"/>

  <!-- ssInfoType -->
  <xs:complexType name="ssInfoType">
    <xs:complexContent>
      <xs:extension base="ssType">
        <xs:attribute name="available" type="xs:boolean" use="required"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- profileType -->
  <xs:complexType name="profileType">
    <xs:sequence>
      <xs:element name="ss" type="ssInfoType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="profile" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="profile" type="profileType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```


Пример 1

Запрос: http://192.168.1.21:9999/commands/ss_domain_profile_info

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_ss_domain_profile_info.xsd">
  <request domain="biysk.local" profile="work_ss"/>
</in>
```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_ss_d
omain_profile_info.xsd">
  <profile name="work_ss" description="for_office">
    <ss
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="
ss.xsd" name="clip" version="5" description="Calling Line Identification Presentation" active="
true" enabled="true" available="true"/>
      <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cnip" version="3" description="Calling Name
Identification Presentation" active="true" enabled="true" available="true">
          <msd>
            <property name="owner" type="atom" description="SS type. Available values:
user | trunk | any" value="any"/>
          </msd>
        </ss>
      <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="chold" version="8" description="Call hold" active="
true" enabled="true" available="true">
          <ssd>
            <property name="dtmf_detector" type="boolean" description="" value="true"/
>
            </ssd>
            <csd>
              <property name="disable_moh" type="boolean" description="Disable music on
hold" value="false"/>
              <property name="dtmf_sequence_as_flash" type="boolean" description="Try
interpret the DTMF sequence as a flash (look at incall feature codes)" value="false"/>
            </csd>
          </ss>
        <ss
          xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="ctr" version="5" description="Call Transfer"
active="true" enabled="true" available="true">
          <msd>
            <property_list name="dependency_list" type="atom" maxlen="0" description="
List of SS that is depended from SS">
              <item value="chold"/>
            </property_list>
          </msd>
          <ssd>
            <property name="dtmf_detector" type="boolean" description="" value="true"/
>
            </ssd>
          </ss>
        <ss

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="conference" version="9" description="Conference
Call, Add-on" active="true" enabled="true" available="true">
    <msd>
        <property name="enumDestroy_mode" type="string" description="Internal
field to validate mode property" value="by_no_master|by_no_more_calls"/>
        <property_list name="conflict_list" type="atom" maxlen="0" description="Li
st of SS that is conflict with current SS">
            <item value="3way"/>
        </property_list>
        <property_list name="dependency_list" type="atom" maxlen="0" description="
List of SS that is depended from SS">
            <item value="chold"/>
            <item value="ctr"/>
        </property_list>
    </msd>
    <csd>
        <property name="destroy_mode" type="enum" description="The mode of
destroying conference (by_no_master|by_no_more_calls)" value="by_no_master"/>
        <property name="max_participants" type="positive_integer" description="The
maximum number of participants in the conference, limited to the number 64 at the top. By
default 16" value="16"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cf_sip" version="9" description="Call Forwarding
by SIP redirection (302 message)" active="true" enabled="true" available="true">
    <msd>
        <property name="owner" type="atom" description="SS type. Available values:
user | trunk | any" value="any"/>
        <property_list name="conflict_list" type="atom" maxlen="0" description="Li
st of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="tsmn_request"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean" description="" value="true"/
>
    </ssd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cc_agent" version="2" description="Contact-Center
agent" active="true" enabled="true" available="true"/>
    <ss

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="m1" version="4" description="Multiline" active="true" enabled="true" available="true">
    <msd>
        <property name="integer_rangeLine_count_out" type="string" description="Internal field" value="[-1,16]"/>
        <property name="integer_rangeLine_count" type="string" description="Internal field" value="[0,16]"/>
    </msd>
    <csd>
        <property name="line_count" type="integer_range" description="Line count" value="3"/>
        <property name="line_count_out" type="integer_range" description="Outgoing lines count" value="-1"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="mcid" version="2" description="Malicious Call Identification" active="true" enabled="true" available="true"/>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="intervention" version="4" description="Intervention" active="true" enabled="true" available="true">
        <msd>
            <property name="enumRestriction" type="string" description="Internal field to validate mode property" value="full_control|implicitly_allowed|explicitly_allowed"/>
        </msd>
        <ssd>
            <property name="dtmf_detector" type="boolean" description="" value="true"/>
        </ssd>
        <csd>
            <property name="restriction" type="enum" description="The restriction to the intervention call (full_control|allowed_only)" value="full_control"/>
        </csd>
    </ss>
</ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="park" version="2" description="Call park" active="true" enabled="true" available="true"/>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="multicast_listen" version="1" description="Multicast Listen Service" active="false" enabled="false" available="false">
        <msd>
            <property name="enumCodec" type="string" description="Internal field to codec property" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|G726-40"/>
        </msd>
    </ss>
</ss>

```

```

        <property name="integer_rangePort" type="string" description="
Internal field to udp port property" value="[10000,65535]"/>
    </msd>
    <csd>
        <property name="ip" type="ip" description="Ip address for
listen multicast."/>
        <property name="port" type="integer_range" description="Port
for listen multicast (10000-65535)."/>
        <property name="codec" type="enum" description="Audio codec
for multicast traffic." value="G722"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="vip_call" version="2" description="Vip call"
active="false" enabled="false" available="false">
    <msd>
        <property name="enumPriority" type="string" description="Call'
s priority" value="0_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
    </msd>
    <csd>
        <property name="priority" type="enum" description="(0_flash_ov
erride | 1_flash | 2_immediate | 3_priority | 4_routine)" value="4_routine"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="voice_page_control" version="2" description="'Voic
e Page Control' service" active="false" enabled="false" available="false">
    <csd>
        <property name="accept_incoming_vp" type="boolean" description="
Accept or Reject voice page calls" value="true"/>
        <property name="block_incoming_vp_by_dnd" type="boolean"
description="Block voice page calls by DND service" value="true"/>
        <property name="mute_incoming_vp" type="boolean" description="
Accept incoming voice page call in silent mode" value="true"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="redial" version="2" description="Last number
redial" active="false" enabled="false" available="false"/>
</ss>
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="fxo_trunk" version="1" description="FX0 trunk
service" active="false" enabled="false" available="false">
    <msd>
        <property name="owner" type="atom" description="SS type.
Available values: user | trunk | any" value="trunk"/>
    </msd>

```

```

        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="sco_black" version="2" description="Selective Call
            Originating, Black list" active="false" enabled="false" available="false">
            <csd>
                <property_list name="blacklist" type="index_address"
                maxlen="50" description="Black list members. Format: <address>[</index>]">
                    <undefined/>
                </property_list>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="zone_page" version="9" description="Zone Page
            call" active="false" enabled="false" available="false">
            <msd>
                <property name="enumPriority" type="string" description="C
                all's priority" value="0_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
                <property name="enumInitiation_policy" type="string"
                description="Initiation call's policy" value="managers|members|any"/>
                <property name="integer_rangeMulticast_port" type="string"
                description="Internal field" value="[1,65535]"/>
                <property name="integer_rangeMulticast_ttl" type="string"
                description="Internal field" value="[1,255]"/>
                <property name="enumMulticast_codec" type="string"
                description="Internal field" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|
                G726-32|G726-40"/>
            </msd>
            <ssd>
                <property name="dtmf_detector" type="boolean" description="
                " value="true"/>
            </ssd>
            <csd>
                <property name="priority" type="enum" description="(0_flas
                h_override | 1_flash | 2_immediate | 3_priority | 4_routine)" value="3_priority"/>
                <property name="intro_pause" type="integer" description="P
                ause before play intro files" value="3"/>
                <property name="announcement_pause" type="integer"
                description="Pause before play announcement files" value="0"/>
                <property name="announcement_playback_cnt" type="integer"
                description="Announcement files playback count" value="1"/>
                <property name="announcement_repeat_time" type="integer"
                description="Announcement files repeat timeout" value="10"/>
                <property name="terminate_when_initiator_disconnects" type="
                boolean" description="Terminate call after initiator disconnected"/>
                <property name="terminate_when_playback_ends" type="boolea
                n" description="Terminate call after playback ends"/>
                <property name="terminate_when_last_listener_leave" type="
                boolean" description="Terminate call after last listener disconnects"/>
            </csd>
        </ss>
    </pre>

```

```

        <property name="initiation_policy" type="enum" description="
Initiation call's policy" value="any"/>
        <property name="display_name" type="string" description="Z
onePage call DisplayName" value="ZonePage"/>
        <property name="play_release_tone" type="boolean"
description="ZonePage release tone" value="false"/>
        <property name="announcement_repeat_digit" type="address"
description="Digit to repeat announcement files" value="#"/>
        <property name="multicast_ip" type="ip" description="Optio
nal multicast address to use for announcement" value="0.0.0.0"/>
        <property name="multicast_port" type="integer_range"
description="Port of multicast address" value="0"/>
        <property name="multicast_codec" type="enum" description="
Codec to be used for multicasting" value="PCMA"/>
        <property name="multicast_ttl" type="integer_range"
description="TTL for multicast" value="1"/>
        <property_list name="members" type="address" maxlen="256"
description="Zone page member list">
            <undefined/>
        </property_list>
        <property_list name="managers" type="address" maxlen="256"
description="Zone page managers list"/>
        <property_list name="intro_files" type="media_resource"
maxlen="10" description="List of Intro media_files"/>
        <property_list name="announcement_files" type="media_resou
rce" maxlen="10" description="List of Announcement media_files"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="meet_me_conference" version="2" description="Meet
Me Conference" active="false" enabled="false" available="false">
    <msd>
        <property name="regimeMode" type="string" description="Int
ernal field to validate mode property" value="master:destroy_mode:room_number|
member:room_number"/>
        <property name="enumDestroy_mode" type="string"
description="Internal field to validate mode property" value="by_no_master|by_no_more_calls"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
        </property_list>
    </msd>
</csd>
    <property name="mode" type="regime" description="The mode
of meet me conference (master|member)"/>
    <property name="destroy_mode" type="enum" description="The
mode of destroying conference (by_no_master|by_no_more_calls)" value="by_no_more_calls"/>

```

```

                                <property name="room_number" type="address" description="R
oom number" value=""/>
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
                                xs:noNamespaceSchemaLocation="ss.xsd" name="teleconference" version="4" description="Teleconfe
                                rence" active="false" enabled="false" available="false">
                                    <msd>
                                        <property name="enumType" type="string" description="Telec
                                onference type: meeting or room" value="meeting|room"/>
                                    </msd>
                                    <csd>
                                        <property name="destroy_by_no_initiator" type="boolean"
                                description="Destroy the teleconference by release from the initiator?" value="true"/>
                                        <property name="template_from_room" type="teleconference_r
                                oom" description="Get template from the teleconference room" value=""/>
                                        <property name="type" type="enum" description="Teleconfere
                                nce type: meeting or room" value="meeting"/>
                                    </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
                                xs:noNamespaceSchemaLocation="ss.xsd" name="assistant" version="2" description="Chief-
                                assistant group" active="false" enabled="false" available="false">
                                    <msd>
                                        <property name="owner" type="atom" description="SS type.
                                Available values: user | trunk | any" value="user"/>
                                    </msd>
                                    <ssd>
                                        <property name="forwarding_ss" type="boolean" description="
                                " value="true"/>
                                    </ssd>
                                    <csd>
                                        <property_list name="assistants" type="index_address"
                                maxlen="4" description="Assistants numbers. Format: <address>[</index>]"/>
                                        <undefined/>
                                    </property_list>
                                        <property_list name="whitelist" type="index_address"
                                maxlen="128" description="White list members. Format: <address>[</index>]"/>
                                        <undefined/>
                                    </property_list>
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
                                xs:noNamespaceSchemaLocation="ss.xsd" name="personal_ivr" version="2" description="Personal
                                IVR script" active="false" enabled="false" available="false">
                                    <msd>

```



```

        <property name="handle_module" type="atom" description=""
value="ss_personal_ivr_handler"/>
        <property name="owner" type="atom" description="SS type.
Available values: user | trunk | any" value="user"/>
        </msd>
        <csd>
            <property name="script" type="ivr_script_id" description="
Script ID" value=""/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="voicemail" version="12" description="Voice mail
service" active="false" enabled="false" available="false">
        <msd>
            <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
                <item value="assistant"/>
                <item value="follow_me"/>
                <item value="cfu_type2"/>
                <item value="meet_me_conference"/>
                <item value="cfu"/>
                <item value="find_me"/>
                <item value="cf_aon"/>
                <item value="cft"/>
            </property_list>
        </msd>
        <ssd>
            <property name="dtmf_detector" type="boolean" description="
" value="true"/>
        </ssd>
        <csd>
            <property name="no_reply" type="boolean" description="Acti
vate service when subscriber no reply" value="true"/>
            <property name="busy" type="boolean" description="Activate
service when subscriber is busy" value="false"/>
            <property name="out_of_service" type="boolean" description="
Activate service when subscriber is out of service" value="true"/>
            <property name="unconditional" type="boolean" description="
Activate service any time" value="false"/>
            <property name="timeout" type="positive_integer"
description="Timeout before call forward (sec)" value="30"/>
            <property name="send_by_email" type="boolean" description="
Send recorded message by email" value="false"/>
            <property name="min_message_duration" type="positive_integ
er" description="The minimum voice mail duration (sec). In case of voice mail duration will be
less that min_message_duration, this message will be dropped" value="0"/>
            <property name="max_message_duration" type="positive_integ
er" description="The maximum voice mail duration (sec). 0 - in case of not limited duration"
value="300"/>

```

```

        <property name="mailbox_volume" type="positive_integer"
description="The mailbox volume (sec)" value="1800"/>
        <property name="override_messages" type="boolean"
description="Override messages in case of mailbox is full" value="true"/>
        <property name="email_from" type="string" description="Ema
il from template. Macro variables: %DATE%, %TIME%, %CALLING%, %DISPLAYNAME%" value="%DISPLAYNA
ME%"/>
        <property name="email_subject" type="string" description="
Email subject template. Macro variables: %DATE%, %TIME%, %CALLING%, %DISPLAYNAME%" value="Voic
email from %CALLING% at %DATE% %TIME%"/>
        <property name="email_body" type="string" description="Ema
il body template. Macro variables: %DATE%, %TIME%, %CALLING%, %DISPLAYNAME%" value=""/>
        <property name="play_message_details" type="boolean"
description="Play message's details (number, time) before message" value="false"/>
        <property name="password" type="address" description="Ther
e is password to have access to the voicemail box" value=""/>
        <property name="max_silence" type="positive_integer"
description="There is time(sec) while service wait for a contiguous period of silence before
terminating an incoming call to voice mail" value="0"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="auto_redial" version="2" description="Auto redial"
    active="false" enabled="false" available="false">
    <csd>
        <property name="recall_timeout" type="positive_integer"
description="Timeout between recalls (sec)" value="15"/>
        <property name="recall_count" type="positive_integer"
description="Recalls' count" value="6"/>
        <property name="no_answer_timeout" type="positive_integer"
description="Awaiting answer timeout (sec)" value="30"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="click_to_call" version="5" description="Click to
call" active="false" enabled="false" available="false">
    <csd>
        <property name="number" type="address" description="User's
number"/>
        <property name="auto_answer" type="boolean" description="A
uto answer" value="true"/>
        <property_list name="hashes" type="string" maxlen="10"
description="User's hashes">
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="distinctive_ring" version="1" description="Destina
tion ringtone change" active="false" enabled="false" available="false">
    <csd>
        <property_list name="filters" type="distinctive_ring"
maxlen="16" description="Distinctive ring for specific numbers">
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cw" version="5" description="Call Waiting" active="
false" enabled="false" available="false">
    <msd>
        <property_list name="dependency_list" type="atom" maxlen="
0" description="List of SS that is depended from SS">
            <item value="chold"/>
        </property_list>
    </msd>
    <ssd>
        <property name="dtmf_detector" type="boolean" description="
" value="true"/>
    </ssd>
</ss>
<ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="call_recording" version="10" description="Call
recording" active="false" enabled="false" available="false">
    <msd>
        <property name="owner" type="atom" description="SS type.
Available values: user | trunk | any" value="any"/>
        <property name="enumMode" type="string" description="Inter
nal field to validate mode property" value="always_on|on_demand|after_answer"/>
    </msd>
    <ssd>
        <property name="locked" type="boolean" description="Can
subscriber manage current SS, or not" value="true"/>
        <property name="dtmf_detector" type="boolean" description="
" value="true"/>
    </ssd>
    <csd>
        <property name="mode" type="enum" description="The mode of
the recording call (always_on|on_demand)"/>
    </csd>
</ss>
<ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="blf" version="3" description="Busy Lamp Field"
active="false" enabled="false" available="false">

```

```

        <csd>
            <property name="max_subscribe" type="positive_integer"
description="Maximum number of subscriptions" value="16"/>
            <property name="ignore_page_calls" type="boolean"
description="Doesn't send BLF notification from Zone Page calls" value="true"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="remote_phone" version="1" description="Remote
phone" active="false" enabled="false" available="false">
        <msd>
            <property name="handle_module" type="atom" description=""
value="ss_remote_phone_handler"/>
        </msd>
        <csd>
            <property name="remote_endpoint" type="interface"
description="Remote interface" value=""/>
            <property name="pin" type="address" description="PIN-code"
/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="presence" version="2" description="SIP Presence"
active="false" enabled="false" available="false">
        <csd>
            <property name="max_subscribe" type="positive_integer"
description="Maximum number of subscriptions" value="16"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="message" version="2" description="Message service"
active="false" enabled="false" available="false">
        <msd>
            <property name="owner" type="atom" description="SS type.
Available values: user | trunk | any" value="any"/>
        </msd>
        <ssd>
            <property name="message_box_size" type="integer"
description="Message box size" value="20"/>
            <property name="remove_only_after_answer" type="boolean"
description="Remove message only after abonent answer" value="false"/>
        </ssd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="ring_back_tone" version="2" description="Ring back
tone" active="false" enabled="false" available="false">

```

```

        <csd>
            <property name="file" type="media_resource" description="File to play on ring"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="voice_page" version="9" description="Voice Page call" active="false" enabled="false" available="false">
        <msd>
            <property name="enumPriority" type="string" description="Call's priority" value="0_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
        </msd>
        <ssd>
            <property name="dtmf_detector" type="boolean" description="DTMF detector" value="true"/>
        </ssd>
        <csd>
            <property name="skip_cf" type="boolean" description="Skipping call forwarding" value="true"/>
            <property name="priority" type="enum" description="(0_flash_override | 1_flash | 2_immediate | 3_priority | 4_routine)" value="3_priority"/>
            <property name="announcement_repeat_time" type="integer" description="Announcement files repeat timeout" value="10"/>
            <property name="announcement_playback_cnt" type="integer" description="Announcement files playback count [0..5]" value="0"/>
            <property name="swap_digit" type="address" description="Digit to swap reception / transmission mode" value="*"/>
            <property name="announcement_repeat_digit" type="address" description="Digit to repeat announcement files" value="#"/>
            <property_list name="intro_files" type="media_resource" maxlen="10" description="List of Intro media_files"/>
            <property_list name="announcement_files" type="media_resource" maxlen="10" description="List of Announcement media_files"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="cfos" version="12" description="Call Forwarding Out of Service" active="false" enabled="false" available="false">
        <msd>
            <property name="owner" type="atom" description="SS type. Available values: user | trunk | any" value="any"/>
            <property_list name="conflict_list" type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                <item value="cfu"/>
                <item value="assistant"/>
                <item value="follow_me"/>
                <item value="tsmn_request"/>
            </property_list>
        </msd>
    </ss>

```

```

        </msd>
        <ssd>
            <property name="forwarding_ss" type="boolean" description="
" value="true"/>
        </ssd>
        <csd>
            <property name="number" type="address" description="Forwar
d number"/>
            <property name="external_number" type="address"
description="External forward number" value=""/>
            <property name="use_external_number" type="boolean"
description="Use external forward number" value="false"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="scr" version="2" description="Selective Call
Rejection" active="false" enabled="false" available="false">
        <csd>
            <property_list name="blacklist" type="index_address"
maxlen="50" description="Black list members. Format: <address>[</index>]"/>
                <undefined/>
            </property_list>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="boss_group" version="1" description="Boss-group"
        active="false" enabled="false" available="false">
        <msd>
            <property name="integer_rangeDelay_ring_timeout" type="str
ing" description="Internal field" value="[0,60]"/>
            <property name="integer_rangeLines_count" type="string"
description="Internal field" value="[0,16]"/>
            <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
                <item value="cw"/>
            </property_list>
            <property_list name="dependency_list" type="atom" maxlen="
0" description="List of SS that is depended from SS">
                <item value="auto_redial"/>
            </property_list>
        </msd>
        <ssd>
            <property name="dtmf_detector" type="boolean" description="
" value="true"/>
        </ssd>
        <csd>
            <property name="display_name" type="string" description="D
isplay name for boss group" value=""/>

```

```

        <property name="delay_ring_timeout" type="integer_range"
description="Timeout before call on delayed members" value="0"/>
        <property name="lines_count" type="integer_range"
description="Incoming boss-group lines count" value="2"/>
        <property name="members_fwd_busy_override" type="boolean"
description="Ignore members CFB for group calls" value="true"/>
        <property name="call_waiting_tone" type="boolean"
description="Play call waiting tone for second(s) calls on group" value="false"/>
        <property_list name="members" type="address" maxlen="8"
description="Members for immediate call">
            <undefined/>
        </property_list>
        <property_list name="delay_members" type="address" maxlen="
8" description="Members for delay call">
            <undefined/>
        </property_list>
        <property_list name="silent_members" type="address" maxlen="
8" description="Members for BLF and pickup only group calls">
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cf_aon" version="8" description="Call Forwarding
AON" active="false" enabled="false" available="false">
    <msd>
        <property name="owner" type="atom" description="SS type.
Available values: user | trunk | any" value="any"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
            <item value="cfu"/>
            <item value="find_me"/>
            <item value="cf_aon"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean" description="
" value="true"/>
    </ssd>
</csd>
    <property name="no_reply" type="boolean" description="Acti
vate service when subscriber no reply" value="false"/>
    <property name="timeout" type="positive_integer"
description="Timeout before call forward (sec)" value="15"/>

```

```

        <property name="number" type="address" description="Forward number"/>
        <property_list name="prefix_number" type="address" maxlen="10" description="Prefix number A">
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="intercom" version="1" description="Intercom"
    active="false" enabled="false" available="false"/>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="direct_call" version="2" description="Direct call"
        active="false" enabled="false" available="false">
            <csd>
                <property name="number" type="address" description="Call number to direct call"/>
                <property name="timeout" type="positive_integer" description="Timeout before direct call (sec)" value="5"/>
            </csd>
        </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="cgg" version="7" description="Group-calling"
        active="false" enabled="false" available="false">
            <msd>
                <property_list name="conflict_list" type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                    <item value="assistant"/>
                    <item value="follow_me"/>
                    <item value="cfu_type2"/>
                    <item value="meet_me_conference"/>
                    <item value="cfu"/>
                    <item value="find_me"/>
                    <item value="cf_aon"/>
                    <item value="cft"/>
                </property_list>
            </msd>
            <ssd>
                <property name="forwarding_ss" type="boolean" description="" value="true"/>
            </ssd>
            <csd>
                <property name="send_call_diversion" type="boolean" description="Send call diversion flag in call signalisation" value="true"/>
                <property_list name="numbers" type="address" maxlen="10" description="Group's numbers">
                    <undefined/>

```



```

        </property_list>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="cfu" version="14" description="Call Forwarding
Unconditional" active="false" enabled="false" available="false">
    <msd>
        <property name="owner" type="atom" description="SS
type. Available values: user | trunk | any" value="any"/>
        <property_list name="conflict_list" type="atom" maxlen="
0" description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address" description="Fo
rward number"/>
        <property name="external_number" type="address"
description="External forward number" value=""/>
        <property name="use_external_number" type="boolean"
description="Use external forward number" value="false"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="find_me" version="7" description="Find Me" active="
false" enabled="false" available="false">
    <msd>
        <property_list name="conflict_list" type="atom" maxlen="
0" description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
            <item value="cfu"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>

```

```

        <property name="send_call_diversion" type="boolean"
description="Send call diversion flag in call signalisation" value="true"/>
        <property_list name="groups" type="find_me_group"
maxlen="32" description="Find me groups"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfu" version="3" description="QSIG: Call
Forwarding Unconditional" active="false" enabled="false" available="false">
    <msd>
        <property name="owner" type="atom" description="SS
type. Available values: user | trunk | any" value="trunk"/>
        <property_list name="conflict_list" type="atom" maxlen="
0" description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="follow_me"/>
        </property_list>
    </msd>
    <csd>
        <property name="hide_forwarding" type="boolean"
description="Hide forwarding fields (rgn, ocdrn)" value="false"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="follow_me" version="11" description="Follow me"
active="false" enabled="false" available="false">
    <msd>
        <property_list name="conflict_list" type="atom" maxlen="
0" description="List of SS that is conflict with current SS">
            <item value="assistant"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address" description="Fo
lloved number" value=""/>
        <property name="pin" type="address" description="PIN-
code" value=""/>
        <property name="external_number" type="address"
description="External forward number" value=""/>
        <property name="use_external_number" type="boolean"
description="Use external forward number" value="false"/>
    </csd>
</ss>
<ss

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="walkie_talkie" version="1" description="Walkie
Talkie" active="false" enabled="false" available="false">
    <ssd>
        <property name="dtmf_detector" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="swap_digit" type="address" description="
Digit to swap reception / transmission mode" value="*"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="multicast_page" version="1" description="Multicast
IP Paging" active="false" enabled="false" available="false">
    <msd>
        <property name="enumCodec" type="string" description="
Internal field" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|G726-40"/>
        <property name="integer_rangePort" type="string"
description="Internal field" value="[10000,65535]"/>
        <property name="integer_rangeTtl" type="string"
description="Internal field" value="[1,255]"/>
    </msd>
    <csd>
        <property name="ip" type="ip" description="IP address
to cast the media to"/>
        <property name="port" type="integer_range" description="
UDP port to cast the media to (10000-65535)"/>
        <property name="codec" type="enum" description="Codec
to be used for multicasting" value="PCMA"/>
        <property name="ttl" type="integer_range" description="
TTL for multicast packets (1-255)" value="1"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cft" version="14" description="Call Forwarding
Unconditional Time Dependent" active="false" enabled="false" available="false">
    <msd>
        <property name="owner" type="atom" description="SS
type. Available values: user | trunk | any" value="any"/>
        <property name="enumTime_format" type="string"
description="Internal field to validate time_format property" value="local|utc"/>
        <property_list name="conflict_list" type="atom" maxlen="
0" description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
        </property_list>
    </msd>
</ss>

```

```

        <item value="cfu"/>
        <item value="find_me"/>
        <item value="cf_aon"/>
    </property_list>
</msd>
<ssd>
    <property name="forwarding_ss" type="boolean"
description="" value="true"/>
</ssd>
<csd>
    <property name="day_1" type="term" description="{{{Fr
omHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}" value="off"/>
    <property name="day_2" type="term" description="{{{Fr
omHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}" value="off"/>
    <property name="day_3" type="term" description="{{{Fr
omHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}" value="off"/>
    <property name="day_4" type="term" description="{{{Fr
omHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}" value="off"/>
    <property name="day_5" type="term" description="{{{Fr
omHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}" value="off"/>
    <property name="day_6" type="term" description="{{{Fr
omHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}" value="off"/>
    <property name="day_7" type="term" description="{{{Fr
omHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}" value="off"/>
    <property name="time_format" type="enum" description="
CFT for Sunday (local | utc)" value="local"/>
</csd>
</ss>
<ss
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="chunt" version="7" description="Call hunt groups"
active="false" enabled="false" available="false">
    <msd>
        <property name="regimeMode" type="string" description="
Internal field to validate search_strategy property" value="group:numbers:queue_strategy|
serial:numbers:cyclic:search_strategy>window_shift_timeout:max_shift_count:queue_strategy|
longest_idle:numbers:cyclic>window_shift_timeout:max_shift_count:queue_strategy|
delay_group:numbers>window_shift_timeout:queue_strategy|manual"/>
        <property name="enumSearch_strategy" type="string"
description="Internal field to validate search_strategy property" value="first|last|random|
longest_idle"/>
        <property name="enumQueue_strategy" type="string"
description="Internal field to validate queue_strategy property" value="wait|drop"/>
        <property_list name="conflict_list" type="atom" maxlen="
0" description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
            <item value="cfu"/>

```

```

        <item value="find_me"/>
        <item value="cf_aon"/>
        <item value="cft"/>
        <item value="cgg"/>
    </property_list>
</msd>
<ssd>
    <property name="delay_retrieve_timeout_1" type="integer" description="Delay retrieve timeout for release group call by group subscriber" value="1000"/>
    <property name="delay_retrieve_timeout_2" type="integer" description="Delay retrieve timeout for release group call by slave subscriber" value="5000"/>
    <property name="delay_retrieve_timeout_3" type="integer" description="Delay retrieve timeout for waiting free group subscriber" value="15000"/>
    <property name="forwarding_ss" type="boolean" description="" value="true"/>
</ssd>
<csd>
    <property name="mode" type="regime" description="The type of the chunt"/>
    <property name="cyclic" type="boolean" description="Flag that is show is call hunt group cyclic or not" value="false"/>
    <property name="reset_window_start_pos_after_call" type="boolean" description="Reset window start position after all incoming calls will be connected with group calls" value="false"/>
    <property name="window_shift_timeout" type="positive_integer" description="Interval after which window shift on the window_shift value" value="0"/>
    <property name="max_shift_count" type="integer" description="The number of times that a window can be shift" value="-1"/>
    <property name="window_size" type="positive_integer" description="The number of simultaneous calls to the call-hunt's subscribers" value="1"/>
    <property name="window_start_pos" type="integer" description="The index (zero-based) of the first subscriber in the window" value="0"/>
    <property name="window_shift" type="positive_integer" description="The value on which shift window_start_pos when window_shift_timeout fired" value="1"/>
    <property name="queue_strategy" type="enum" description="(wait | drop)" value="wait"/>
    <property name="search_strategy" type="enum" description="(first | last | random | longest_idle)" value="first"/>
    <property name="queue_size" type="positive_integer" description="Wait calls' queue size" value="0"/>
    <property_list name="numbers" type="address" maxlen="10" description="Call-hunt member list">
        <undefined/>
    </property_list>
</csd>
</ss>
<ss>

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cfnr_type2" version="11" description="Call
Forwarding No Reply without notification" active="false" enabled="false" available="false">
    <msd>
        <property name="owner" type="atom" description="SS
type. Available values: user | trunk | any" value="any"/>
        <property_list name="conflict_list" type="atom" maxlen="
0" description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="cfnr"/>
            <item value="follow_me"/>
            <item value="tsmn_request"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address" description="Fo
rward number"/>
        <property name="timeout" type="positive_integer"
description="Timeout before call forward (sec)" value="15"/>
        <property name="use_redirection_counter" type="boolean"
description="Use redirectionCounter field to count hops" value="false"/>
        <property name="external_number" type="address"
description="External forward number" value=""/>
        <property name="use_external_number" type="boolean"
description="Use external forward number" value="false"/>
    </csd>
</ss>
<ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="acb" version="2" description="Anonymous Call
Blocking" active="false" enabled="false" available="false"/>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cfu_type2" version="14" description="Call
Forwarding Unconditional without notification" active="false" enabled="false" available="false"
">
        <msd>
            <property name="owner" type="atom" description="SS
type. Available values: user | trunk | any" value="any"/>
            <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
                <item value="assistant"/>
                <item value="follow_me"/>
            </property_list>
        </msd>

```

```

        <ssd>
            <property name="forwarding_ss" type="boolean"
description="" value="true"/>
        </ssd>
        <csd>
            <property name="number" type="address" description="
Forward number"/>
            <property name="use_redirection_counter" type="boo
lean" description="Use redirectionCounter field to count hops" value="false"/>
            <property name="external_number" type="address"
description="External forward number" value=""/>
            <property name="use_external_number" type="boolean
" description="Use external forward number" value="false"/>
        </csd>
    </ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cfnr" version="11" description="Call Forwarding No
Reply" active="false" enabled="false" available="false">
    <msd>
        <property name="owner" type="atom" description="SS
type. Available values: user | trunk | any" value="any"/>
        <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="tsmn_request"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address" description="
Forward number"/>
        <property name="external_number" type="address"
description="External forward number" value=""/>
        <property name="use_external_number" type="boolean
" description="Use external forward number" value="false"/>
        <property name="timeout" type="positive_integer"
description="Timeout before call forward (sec)" value="15"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="sca" version="2" description="Selective Call
Acceptance" active="false" enabled="false" available="false">
    <csd>

```

```

        <property_list name="whitelist" type="index_addresses" maxlen="50" description="White list members. Format: <address>[</index>]"/>
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="rbp" version="3" description="SS for restricting
call-processes by NI and PIN" active="false" enabled="false" available="false">
    <csd>
        <property name="implicit_call" type="boolean"
description="Use implicit RBP call (Request PIN-code in case of use restricted call)" value="false"/>
        <property_list name="restricted_ni" type="ni"
maxlen="6" description="Restricted number indication">
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="cliro" version="2" description="Calling Line
Identification Restriction Override" active="false" enabled="false" available="false"/>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="alarm" version="8" description="Alarm
Call" active="false" enabled="false" available="false">
        <msd>
            <property name="handle_module" type="atom"
description="" value="ss_alarm_handler"/>
        </msd>
        <ssd>
            <property name="collect_digits_timeout" type="
integer" description="Wait timeout to collect cancel digits" value="60"/>
        </ssd>
        <csd>
            <property name="no_answer_timeout" type="integ
er" description="Awaiting answer timer (value in seconds)" value="45"/>
            <property name="call_attempt_count" type="inte
ger" description="The count of the call's attempt" value="3"/>
            <property name="call_attempt_timeout" type="in
teger" description="Awaiting timer between calls' attempts" value="60"/>
            <property name="cancel_digits" type="address"
description="Digits to cancel alarm" value="*"/>
            <property name="alarm_1" type="term"
description="'{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]"}" value="off"/>
    </ss>
</ss>

```



```

                <property name="alarm_2" type="term"
description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]"}"" value="off"/>
                <property name="alarm_3" type="term"
description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]"}"" value="off"/>
                <property name="alarm_4" type="term"
description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]"}"" value="off"/>
                <property name="alarm_5" type="term"
description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]"}"" value="off"/>
                <property name="alarm_6" type="term"
description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]"}"" value="off"/>
                <property name="alarm_7" type="term"
description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]"}"" value="off"/>
                <property name="alarm_8" type="term"
description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]"}"" value="off"/>
                <property name="alarm_9" type="term"
description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]"}"" value="off"/>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="speed_dial" version="5" description="Spe
ed dial (Abbreviated dialing)" active="false" enabled="false" available="false">
            <msd>
                <property name="enumType" type="string"
description="The type of address book. address_book_10 supports 10 number in the address book.
address_book_100 supports 100 numbers on the address book" value="address_book_10|
address_book_100"/>
            </msd>
            <csd>
                <property name="type" type="enum" description="
(address_book_10 | address_book_100)" value="address_book_100"/>
                <property_list name="map" type="speed_dial_ite
m" maxlen="50" description="Map between short dial and full address. Format: <short_dial>/
<address>"/>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="fax_to_email" version="2" description="F
ax to email" active="false" enabled="false" available="false">
            <csd>

```

```

        <property name="email" type="string"
description="Email address" value=""/>
        <property name="send_error_report" type="boolean" description="Send email in case of fax received failed" value="false"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="flexicall" version="1" description="FlexiCall" active="false" enabled="false" available="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available values: user | trunk | any." value="user"/>
        <property name="enumMode" type="string"
description="Internal field to validate mode property" value="all|external|internal"/>
        <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
            <item value="cgg"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="mode" type="enum" description="Mode operating. Available values (all | external | internal)"/>
        <property_list name="external_numbers" type="address" maxlen="10" description="Group's external numbers"/>
        <property_list name="internal_numbers" type="address" maxlen="10" description="Group's internal numbers"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="auto_redial_with_callback" version="5"
description="Auto redial with callback" active="false" enabled="false" available="false">
    <msd>
        <property name="enumRingback" type="string"
description="Ringback" value="tone|moh"/>
    </msd>
    <csd>
        <property name="recall_timer" type="positive_integer" description="Time between recalls (sec)" value="90"/>
        <property name="recall_count" type="positive_integer" description="Recalls' count" value="30"/>
        <property name="a_no_answer_timeout" type="positive_integer" description="Awaiting B-side answer timeout (sec)" value="10"/>
        <property name="b_no_answer_timeout" type="positive_integer" description="Awaiting A-side answer timeout (sec)" value="30"/>
    </csd>
</ss>

```

```

                <property name="ringback" type="enum"
description="Ringback (tone | moh)" value="tone"/>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="callback" version="5" description="CallB
ack" active="false" enabled="false" available="false">
            <ssd>
                <property name="collect_digits_timeout" type="
integer" description="Wait timeout to collect cancel digits" value="60"/>
                <property name="a_onhook_timeout" type="positi
ve_integer" description="Timeout for A subscriber onhook (sec)" value="5"/>
                <property name="max_call_retry" type="positive
_integer" description="The maximum count of call's retry to initial subscriber" value="5"/>
                <property name="dtmf_detector" type="boolean"
description="" value="true"/>
            </ssd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="3way" version="6" description="3-Way
Conference" active="false" enabled="false" available="false">
            <msd>
                <property_list name="dependency_list" type="atom"
maxlen="0" description="List of SS that is depended from SS">
                    <item value="chold"/>
                </property_list>
                <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
                    <item value="conference"/>
                </property_list>
            </msd>
            <ssd>
                <property name="dtmf_detector" type="boolean"
description="" value="true"/>
            </ssd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="introduce" version="1" description="Intr
oduce announcement from callee" active="false" enabled="false" available="false">
            <csd>
                <property name="announcement" type="media_reso
urce" description="Announcement media file"/>
            </csd>
        </ss>
    </ss>

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="distinctive_picture" version="2"
description="Display caller image by groups" active="false" enabled="false" available="false">
    <csd>
        <property_list name="groups" type="distinctive
_picture" maxlen="20" description="Distinctive picture for specific numbers"/>
    </csd>
</ss>
<ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="find_me_no_response" version="6"
description="Find Me no response" active="false" enabled="false" available="false">
    <msd>
        <property_list name="conflict_list" type="atom
" maxlen="0" description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="ctu"/>
            <item value="cgg"/>
            <item value="follow_me"/>
            <item value="follow_me_no_response"/>
            <item value="find_me"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="false"/>
    </ssd>
    <csd>
        <property name="send_call_diversion" type="boo
lean" description="Send call diversion flag in call signalisation" value="true"/>
        <property name="timeout" type="positive_intege
r" description="Timeout before call forward (sec)" value="20"/>
        <property_list name="groups" type="find_me_gro
up" maxlen="32" description="Find me groups"/>
    </csd>
</ss>
<ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="follow_me_no_response" version="8"
description="Follow me no response" active="false" enabled="false" available="false">
    <msd>
        <property_list name="conflict_list" type="atom
" maxlen="0" description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="cft"/>
            <item value="cgg"/>
            <item value="follow_me"/>
        </property_list>

```

```

</msd>
<ssd>
  <property name="forwarding_ss" type="boolean"
description="" value="true"/>
</ssd>
<csd>
  <property name="number" type="address"
description="Followed number" value=""/>
  <property name="timeout" type="positive_intege
r" description="Timeout before call forward (sec)" value="20"/>
  <property name="pin" type="address"
description="PIN-code" value=""/>
  <property name="external_number" type="address
" description="External forward number" value=""/>
  <property name="use_external_number" type="boo
lean" description="Use external forward number" value="false"/>
</csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="sco_white" version="2" description="Sele
ctive Call Originating, White list" active="false" enabled="false" available="false">
  <csd>
    <property_list name="whitelist" type="index_ad
dress" maxlen="50" description="White list members. Format: <address>[/<index>]"/>
    <undefined/>
  </property_list>
  </csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="tsmn_request" version="3" description="S
upport request on TSMN" active="false" enabled="false" available="false">
  <msd>
    <property name="owner" type="atom" description="
SS type. Available values: user | trunk | any" value="trunk"/>
  </msd>
  <csd>
    <property name="timeout" type="positive_intege
r" description="Timeout on TSMN request(s) (msec)" value="3000"/>
    <property name="tsmn_trunk" type="trunk"
description="Trunk to TSMN server"/>
    <property name="tsmn_trunk_backup" type="trunk
" description="Backup trunk to TSMN server" value=""/>
  </csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="dnd" version="3" description="Do not
Disturb" active="false" enabled="false" available="false">

```

```

        <csd>
            <property_list name="whitelist" type="index_ad
dress" maxlen="10" description="White list members. Format: <address>[</index>]">
                <undefined/>
            </property_list>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="rfc" version="2" description="Rejection
of Forwarded Calls" active="false" enabled="false" available="false"/>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="fbc" version="1" description="Forwarding
Barring Call" active="false" enabled="false" available="false">
                <msd>
                    <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
                </msd>
            </ss>
            <ss
                xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfnr" version="2" description="QSIG
: Call Forwarding No Reply" active="false" enabled="false" available="false">
                    <msd>
                        <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="trunk"/>
                        <property_list name="conflict_list" type="
atom" maxlen="0" description="List of SS that is conflict with current SS">
                            <item value="cfu"/>
                        </property_list>
                    </msd>
                </ss>
                <ss
                    xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfb" version="11" description="Call
Forwarding Busy" active="false" enabled="false" available="false">
                        <msd>
                            <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
                            <property_list name="conflict_list" type="
atom" maxlen="0" description="List of SS that is conflict with current SS">
                                <item value="cfu"/>
                                <item value="assistant"/>
                                <item value="follow_me"/>
                                <item value="tsmn_request"/>
                            </property_list>
                        </msd>
                    </ss>
                </ss>
            </ss>
        </ss>
    </ss>

```

```

an" description="" value="true"/>
                                <property name="forwarding_ss" type="boole
                                </ssd>
                                <csd>
                                <property name="number" type="address"
description="Forward number"/>
                                <property name="external_number" type="add
ress" description="External forward number" value=""/>
                                <property name="use_external_number" type="
boolean" description="Use external forward number" value="false"/>
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="avatar" version="2" description="Picture
for outgoing calls" active="false" enabled="false" available="false">
                                <csd>
                                <property name="pic_addr" type="uri_string
" description="URL to access the image" value=""/>
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfb" version="3" description="QSIG:
Call Forwarding Busy" active="false" enabled="false" available="false">
                                <msd>
                                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="trunk"/>
                                <property_list name="conflict_list" type="
atom" maxlen="0" description="List of SS that is conflict with current SS">
                                    <item value="cfu"/>
                                </property_list>
                                </msd>
                                <csd>
                                <property name="hide_forwarding" type="boo
lean" description="Hide forwarding fields (rgn, ocdrn)" value="false"/>
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="fax_receiver" version="2" description="E
nable fax receiving ability for terminals which is not support fax by their own" active="false
" enabled="false" available="false"/>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="clir" version="3" description="
Calling Line Identification Restriction" active="false" enabled="false" available="false">
                                <msd>
                                <property name="enumMode" type="string
" description="Internal field to validate mode property" value="user_defined|restrict_all"/>

```

```

        </msd>
        <csd>
            <property name="mode" type="enum"
description="The mode of the working clir service (user_defined|restrict_all)" value="restrict
_all"/>
        </csd>
    </ss>
    <ss
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="teleconference_manager"
version="3" description="Teleconference manager" active="false" enabled="false" available="fal
se">
        <csd>
            <property_list name="second_line" type="
address" maxlen="10" description="Second line numbers"/>
        </csd>
    </ss>
    <ss
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="pickup" version="2"
description="Call Pickup" active="false" enabled="false" available="false">
        <csd>
            <property_list name="pickup_groups"
type="index_string" maxlen="10" description="Call pick-up's groups names. Format: <Name>[<Inx
]>
            <undefined/>
        </property_list>
    </csd>
</ss>
    <ss
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cpip" version="2" description="
Calling Picture Identification Presentation" active="false" enabled="false" available="false">
        <msd>
            <property name="enumDisplay_mode" type="
string" description="Internal field to validate 'display-mode' property" value="icon|
fullscreen"/>
        </msd>
        <csd>
            <property name="display_mode" type="en
um" description="Sets the picture display mode: icon mode or fullscreen mode." value="icon"/>
        </csd>
    </ss>
</profile>
</out>

```


Нс_ss_domain_profile_list

Команда позволяет просмотреть список доменных профилей услуг.

Аналог команды в CoCon:

/domain/<DOMAIN>/ss/profile/info

Шаблон http запроса:

http://host:port/commands/ss_domain_profile_list

Код http ответа:

200 – в случае успеха.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="ss.xsd"/>
  <xs:include schemaLocation="error.xsd"/>

  <!-- profileType -->
  <xs:complexType name="profileType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="profile" type="profileType" minOccurs="0" maxOccurs="unbounded"
        />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1

Запрос: http://192.168.1.21:9999/commands/ss_domain_profile_list

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_ss_domain_profile_list.xsd">
  <request domain="biysk.local"/>
</in>

```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ss_domain_profile_list.xsd">
  <profile name="work_ss" description="for_office"/>
  <profile name="min_ss" description="only_clip_cnip_chold"/>
  <profile name="ss_test" description="test"/>
</out>
```

[Hc_ss_domain_profile_remove](#)

Команда позволяет удалить несколько системных профилей по имени.

Аналог команды в CoCon:

`/domain/<DOMAIN>/ss/profile/remove`

Шаблон http запроса:

`http://host:port/commands/ss_domain_profile_remove`

Код http ответа:

204 – в случае успеха;

200 – в случае ошибки (с указанием ошибки в теле ответа)

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>

  <!-- removeProfileType -->

  <xs:complexType name="removeProfileType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- requestType -->

  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="profile" type="removeProfileType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->

  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->

  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Пример 1

Запрос: http://192.168.1.21:9999/commands/ss_domain_profile_remove

```
<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_ss_domain_profile_remove.xsd">
  <request domain="biysk.local">
    <profile name="ss_test"/>
    <profile name="min_ss_copy"/>
  </request>
</in>
```

Ответ:

204

Пример 2

В случае, если профили с именем profile_test не существует

Запрос: http://192.168.1.21:9999/commands/ss_domain_profile_remove

```
<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_ss_domain_profile_remove.xsd">
  <request domain="biysk.local">
    <profile name="ss_test"/>
    <profile name="min_ss_copy"/>
  </request>
</in>
```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ss_domain_profile_remove.xsd">
  <error cmd="hc_ss_domain_profile_remove" reason="not_exists" body="not_exists"
  entity="profile_test"/>
</out>
```

[Hc_ss_domain_profile_set](#)

Команда позволяет создать новые и/или изменить существующие доменные профили услуг.

Аналог команды в CoCon:

`/domain/<DOMAIN>/ss/profile/set`

Шаблон http запроса:

http://host:port/commands/ss_domain_profile_set

Код http ответа:

201 – в случае успеха;

200 – в случае ошибки (с указанием ошибки в теле ответа).

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>

  <!-- propertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- ssType -->
  <xs:complexType name="ssType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="enable" type="xs:boolean" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
  </xs:complexType>

  <!-- setProfileType -->
  <xs:complexType name="setProfileType">
    <xs:sequence>
      <xs:element name="ss" type="ssType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="profile" type="setProfileType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->

```

```

<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1

Запрос:

http://192.168.1.21:9999/commands/ss_domain_profile_set

```

<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_ss_domain_profile_set.xsd">
  <request domain="biysk.local">
    <profile name="ss_test" description="description 1">
      <ss name="chold" enable="true" active="false"/>
      <ss name="cfu" enable="false" active="false"/>
      <ss name="cfnr" enable="true" active="true">
        <property value="5" key="timeout"/>
        <property value="235" key="number"/>
      </ss>
    </profile>
    <profile name="min_ss_copy" description="description 2">
      <ss name="chold" enable="true" active="false"/>
    </profile>
  </request>
</in>

```

Ответ:

201

Пример 2

В случае, если при изменения настроек профиля с именем profile1 произошла ошибка (при этом profile2 успешно изменен):

Запрос: http://192.168.1.21:9999/commands/ss_domain_profile_set


```

<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_ss_domain_profile_set.xsd">
  <request domain="biysk.local">
    <profile name="ss_test" description="description 1">
      <ss name="chold" enable="true" active="false"/>
      <ss name="cfu" enable="false" active="false"/>
      <ss name="cfnr" enable="true" active="true">
        <property value="5" key="timeout"/>
      </ss>
    </profile>
    <profile name="profile_hold" description="description 2">
      <ss name="chold" enable="true" active="false"/>
    </profile>
  </request>
</in>

```

Ответ:

200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ss_domain_profile_set.xsd">
  <error cmd="hc_ss_domain_profile_set" reason="undefined_properties"
  body="{undefined_properties, [{cfnr, [number]}}" entity="ss_test"/>
</out>

```

Hc_ss_feature_code - Команды управления Feature кодами для работы с услугами

- [Hc_ss_feature_code_clean](#)
- [Hc_ss_feature_code_info](#)
- [Hc_ss_feature_code_set](#)
- [Hc_ss_feature_code_properties_info](#)

[Hc_ss_feature_code_clean](#)

Команда позволяет восстановить значение FeatureCode-а для работы с услугами в "Значение по умолчанию" в рамках определенного домена.

Аналог команды в CoCon:

domain/<DOMAIN>/ss/feature-codes/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_feature_code_clean

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Reset supplementary services' feature codes for specific domain.
      Usage: http://server:9999/commands/ss\_feature\_code\_clean
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <!-- non-empty token -->
  <xs:simpleType name="prefixType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
  <!-- featureCodeType -->
  <xs:complexType name="featureCodeType">
    <xs:attribute name="ss_name" type="xs:string" use="required"/>
    <xs:attribute name="feature_key" type="xs:string" use="required"/>
    <xs:attribute name="feature" type="xs:string" use="optional"/>
    <xs:attribute name="prefix" type="prefixType" use="optional"/>
    <xs:attribute name="default_prefix" type="prefixType" use="optional"/>
    <xs:attribute name="usage_samples" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- okType -->
  <xs:complexType name="okType"/>
  <!-- duplicatesType -->
  <xs:complexType name="duplicatesType">
    <xs:sequence>
      <xs:element name="feature_code" type="featureCodeType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="feature_code" type="featureCodeType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

<!-- out-->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="result">
        <xs:complexType>
          <xs:choice>
            <xs:element name="ok" type="okType"/>
            <xs:element name="duplicates" type="duplicatesType"/>
          </xs:choice>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1

Запрос:

http://192.168.1.21:9999/commands/ss_feature_code_clean

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local">
    <feature_code ss_name="cfu" feature_key="activate"/>
    <feature_code ss_name="cfu" feature_key="activate_pulse"/>
    <feature_code ss_name="cfb" feature_key="test_number_pulse"/>
  </request>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_feature_code_clean.xsd">
  <result>
    <duplicates>
      <feature_code ss_name="cfb" feature_key="deactivate_pulse" feature="deactivate"
        prefix="#222" default_prefix="#222" usage_samples="" description="Deactivate 'Call Forwarding
        Busy' (pulse)"/>
      <feature_code ss_name="cfu" feature_key="deactivate_pulse" feature="deactivate"
        prefix="#222" default_prefix="#222" usage_samples="" description="Deactivate 'Call Forwarding
        Unconditional' (pulse)"/>
    </duplicates>
  </result>
</out>
```

Пример 2

Восстановить все значения по-умолчанию

Запрос:

http://192.168.1.21:9999/commands/ss_feature_code_clean

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local">
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_feature_code_clean.xsd">
  <result>
    <ok/>
  </result>
</out>
```

[Hc_ss_feature_code_info](#)

Команда позволяет получить список Feature кодов для работы с услугами в рамках определенного домена.

Аналог команды в CoCon:

domain/<DOMAIN>/ss/feature-codes/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_feature_code_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get supplementary services' feature codes for specific domain.
      Usage: http://server:9999/commands/ss\_feature\_code\_info
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <!-- non-empty token -->
  <xs:simpleType name="prefixType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
  <!-- featureCodeType -->
  <xs:complexType name="featureCodeType">
    <xs:attribute name="ss_name" type="xs:string" use="required"/>
    <xs:attribute name="feature_key" type="xs:string" use="required"/>
    <xs:attribute name="feature" type="xs:string" use="required"/>
    <xs:attribute name="prefix" type="prefixType" use="required"/>
    <xs:attribute name="default_prefix" type="prefixType" use="required"/>
    <xs:attribute name="usage_samples" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="result">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="feature_code" type="featureCodeType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```
</xs:complexType>  
</xs:element>  
</xs:schema>
```

Пример

Запрос:

http://192.168.1.21:9999/commands/ss_feature_code_info

```
<?xml version="1.0"?>  
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">  
  <request domain="biysk.local"/>  
</in>
```

Ответ: 200


```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="ss_feature_code_info.xsd">
  <result>
    <feature_code ss_name="cf_aon" feature_key="activate" feature="activate" prefix="*20"
      default_prefix="*20" usage_samples="CODE PREFIX_NUMBER NUMBER;1 50 5000" description="Activate
      'Call Forwarding AON'"/>
    <feature_code ss_name="cfu" feature_key="activate" feature="activate" prefix="*21"
      default_prefix="*21" usage_samples="NUMBER;MODE NUMBER;5000;1 5000" description="Activate
      'Call Forwarding Unconditional'"/>
    <feature_code ss_name="cfb" feature_key="activate" feature="activate" prefix="*22"
      default_prefix="*22" usage_samples="MODE NUMBER;NUMBER;1 5000;5000" description="Activate
      'Call Forwarding Busy'"/>
    <feature_code ss_name="cfos" feature_key="activate" feature="activate" prefix="*24"
      default_prefix="*24" usage_samples="NUMBER;MODE NUMBER;5000;1 5000" description="Activate
      'Call Forwarding Out of Service'"/>
    <feature_code ss_name="meet_me_conference" feature_key="create_room" feature="teleconf
      erence" prefix="*270" default_prefix="*270" usage_samples="MEET_ME_ROOM;1000" description="Cre
      ate the meet me conference room"/>
    <feature_code ss_name="meet_me_conference" feature_key="enter_room" feature="teleconfe
      rence" prefix="*271" default_prefix="*271" usage_samples="MEET_ME_ROOM;1000" description="Ente
      r the meet me conference room"/>
    <feature_code ss_name="cfnr" feature_key="activate" feature="activate" prefix="*27"
      default_prefix="*27" usage_samples="NUMBER;MODE NUMBER;5000;1 5000" description="Activate
      'Call Forwarding No Reply'"/>
    <feature_code ss_name="cft" feature_key="activate" feature="activate" prefix="*28"
      default_prefix="*28" usage_samples="DAYS HH1MM1HH2MM2 ... NUMBER;DAYS HH1MM1HH2MM2 ... MODE
      NUMBER;12345 00000859 12001259 18002359 500012345 00000859 12001259 18002359 1 5000"
      description="Activate 'Call Forwarding Unconditional Time Dependent'"/>
    <feature_code ss_name="follow_me" feature_key="activate" feature="activate" prefix="*2
      3" default_prefix="*23" usage_samples=";PIN; LOCAL_NUMBER;PIN LOCAL_NUMBER;PIN MODE
      LOCAL_NUMBER;1234; 5000;1234 5000;1234 1 5000" description="Activate 'Follow me'"/>
    <feature_code ss_name="follow_me_no_response" feature_key="activate" feature="activate
      " prefix="*25" default_prefix="*25" usage_samples=";PIN; LOCAL_NUMBER;PIN LOCAL_NUMBER;PIN
      MODE LOCAL_NUMBER;1234; 5000;1234 5000;1234 1 5000" description="Activate 'Follow me No
      Response'"/>
    <feature_code ss_name="dnd" feature_key="activate" feature="activate" prefix="*26"
      default_prefix="*26" usage_samples="" description="Activate 'Do not Disturb'"/>
    <feature_code ss_name="PIN" feature_key="activate" feature="activate" prefix="*29"
      default_prefix="*29" usage_samples="OLD_PIN NEW_PIN NEW_PIN;1111 1234 1234" description="Chang
      e PIN code"/>
    <feature_code ss_name="cf_aon" feature_key="trigger" feature="trigger" prefix="*D20"
      default_prefix="*D20" usage_samples="CODE PREFIX_NUMBER NUMBER;1 50 5000" description="Activat
      e/Deactivate 'Call Forwarding AON' in trigger mode"/>
    <feature_code ss_name="cfu" feature_key="trigger" feature="trigger" prefix="*D21"
      default_prefix="*D21" usage_samples="NUMBER;MODE NUMBER;5000;1 5000" description="Activate/
      Deactivate 'Call Forwarding Unconditional' in trigger mode"/>
  </result>
</out>

```

```

    <feature_code ss_name="cfb" feature_key="trigger" feature="trigger" prefix="*D22"
default_prefix="*D22" usage_samples="NUMBER;5000" description="Activate/Deactivate 'Call
Forwarding Busy' in trigger mode"/>
    <feature_code ss_name="cfos" feature_key="trigger" feature="trigger" prefix="*D24"
default_prefix="*D24" usage_samples="NUMBER;MODE NUMBER;5000;1 5000" description="Activate/
Deactivate 'Call Forwarding Out of Service' in trigger mode"/>
    <feature_code ss_name="cfnr" feature_key="trigger" feature="trigger" prefix="*D27"
default_prefix="*D27" usage_samples="NUMBER;MODE NUMBER;5000;1 5000" description="Activate
'Call Forwarding No Reply' in trigger mode"/>
    <feature_code ss_name="cft" feature_key="trigger" feature="trigger" prefix="*D28"
default_prefix="*D28" usage_samples="DAYS HH1MM1HH2MM2 ... NUMBER;DAYS HH1MM1HH2MM2 ... MODE
NUMBER;12345 00000859 12001259 18002359 500012345 00000859 12001259 18002359 1 5000"
description="Activate/Deactivate 'Call Forwarding Unconditional Time Dependent' in trigger
mode"/>
    <feature_code ss_name="dnd" feature_key="trigger" feature="trigger" prefix="*D26"
default_prefix="*D26" usage_samples="" description="Activate/Deactivate 'Do not Disturb' in
trigger mode"/>
    <feature_code ss_name="cfu_type2" feature_key="trigger" feature="trigger" prefix="*D41
" default_prefix="*D41" usage_samples="NUMBER;MODE NUMBER;5000;1 5000" description="ActivateDe
activate 'Call Forwarding Unconditional without notification' in trigger mode"/>
    <feature_code ss_name="cfnr_type2" feature_key="trigger" feature="trigger" prefix="*D4
2" default_prefix="*D42" usage_samples="NUMBER;MODE NUMBER;5000;1 5000" description="Activate/
Deactivate 'Call Forwarding No Reply without notification' in trigger mode"/>
    <feature_code ss_name="cw" feature_key="trigger" feature="trigger" prefix="*D43"
default_prefix="*D43" usage_samples="" description="Activate/Deactivate 'Call Waiting' in
trigger mode"/>
    <feature_code ss_name="redial" feature_key="trigger" feature="trigger" prefix="*D77"
default_prefix="*D77" usage_samples="" description="Activate/Deactivate 'Last number redial'
in trigger mode"/>
    <feature_code ss_name="direct_call" feature_key="trigger" feature="trigger" prefix="*D
53" default_prefix="*D53" usage_samples="NUMBER;5000" description="Activate/Deactivate 'Direct
call' in trigger mode"/>
    <feature_code ss_name="alarm" feature_key="trigger" feature="trigger" prefix="*D55"
default_prefix="*D55" usage_samples="ARM_NUMBER DAY HHMM;ARM_NUMBER HHMM;1 5 0900;1 0900"
description="Activate/Deactivate one-time alarm in trigger mode"/>
    <feature_code ss_name="alarm" feature_key="trigger" feature="trigger" prefix="*D56"
default_prefix="*D55" usage_samples="ARM_NUMBER DAY HHMM;ARM_NUMBER HHMM;1 5 0900;1 0900"
description="Activate/Deactivate one-time alarm in trigger mode"/>
    <feature_code ss_name="rfc" feature_key="trigger" feature="trigger" prefix="*D64"
default_prefix="*D64" usage_samples="" description="Activate/Deactivate 'Rejection of
Forwarded Calls' in trigger mode"/>
    <feature_code ss_name="acb" feature_key="trigger" feature="trigger" prefix="*D16"
default_prefix="*D16" usage_samples="" description="Activate/Deactivate 'Anonymous Call
Blocking' in trigger mode"/>
    <feature_code ss_name="chold" feature_key="trigger" feature="trigger" prefix="*D94"
default_prefix="*D94" usage_samples="" description="Activate/Deactivate 'Call hold' in trigger
mode"/>
    <feature_code ss_name="ctr" feature_key="trigger" feature="trigger" prefix="*D96"
default_prefix="*D96" usage_samples="" description="Activate/Deactivate 'Call Transfer' in
trigger mode"/>

```

```

    <feature_code ss_name="3way" feature_key="trigger" feature="trigger" prefix="*D95"
default_prefix="*D95" usage_samples="" description="Activate/Deactivate '3-Way Conference' in
trigger mode"/>
    <feature_code ss_name="clir" feature_key="trigger" feature="trigger" prefix="*D31"
default_prefix="*D31" usage_samples="" description="Activate/Deactivate 'Calling Line
Identification Restriction' in trigger mode"/>
    <feature_code ss_name="cf_aon" feature_key="test" feature="test" prefix="*#20"
default_prefix="*#20" usage_samples=";CODE PREFIX_NUMBER NUMBER;1 50 5000" description="Check
'Call Forwarding AON'"/>
    <feature_code ss_name="cfu" feature_key="test" feature="test" prefix="*#21"
default_prefix="*#21" usage_samples=";NUMBER;5000" description="Check 'Call Forwarding
Unconditional'"/>
    <feature_code ss_name="cfb" feature_key="test" feature="test" prefix="*#22"
default_prefix="*#22" usage_samples=";NUMBER;5000" description="Check 'Call Forwarding Busy'"/
>
    <feature_code ss_name="cfos" feature_key="test" feature="test" prefix="*#24"
default_prefix="*#24" usage_samples=";NUMBER;5000" description="Check 'Call Forwarding Out of
Service'"/>
    <feature_code ss_name="cfnr" feature_key="test" feature="test" prefix="*#27"
default_prefix="*#27" usage_samples="NUMBER;5000" description="Check 'Call Forwarding No
Reply'"/>
    <feature_code ss_name="cft" feature_key="test" feature="test" prefix="*#28"
default_prefix="*#28" usage_samples=";DAY;1" description="Check 'Call Forwarding Unconditional
Time Dependent'"/>
    <feature_code ss_name="follow_me" feature_key="test" feature="test" prefix="*#23"
default_prefix="*#23" usage_samples="" description="Check 'Follow me'"/>
    <feature_code ss_name="follow_me_no_response" feature_key="test" feature="test" prefix="
*#25" default_prefix="*#25" usage_samples="" description="Check 'Follow me No Response'"/>
    <feature_code ss_name="dnd" feature_key="test" feature="test" prefix="*#26"
default_prefix="*#26" usage_samples="" description="Check 'Do not Disturb'"/>
    <feature_code ss_name="cfu_type2" feature_key="test" feature="test" prefix="*#41"
default_prefix="*#41" usage_samples="" description="Check 'Call Forwarding Unconditional
without notification'"/>
    <feature_code ss_name="cfnr_type2" feature_key="test" feature="test" prefix="*#42"
default_prefix="*#42" usage_samples="" description="Check 'Call Forwarding No Reply without
notification'"/>
    <feature_code ss_name="cw" feature_key="test" feature="test" prefix="*#43"
default_prefix="*#43" usage_samples="" description="Check 'Call Waiting'"/>
    <feature_code ss_name="follow_me_no_response" feature_key="test_timeout" feature="test
" prefix="*#125" default_prefix="*#125" usage_samples="" description="Check No Response
timeout for 'Follow me No Response'"/>
    <feature_code ss_name="acb" feature_key="test" feature="test" prefix="*#16"
default_prefix="*#16" usage_samples="" description="Check 'Anonymous Call Blocking'"/>
    <feature_code ss_name="privacy" feature_key="test" feature="test" prefix="*#186"
default_prefix="*#186" usage_samples="" description="Check 'Privacy' mode"/>
    <feature_code ss_name="auto_redial_with_callback" feature_key="stop" feature="use"
prefix="*#39" default_prefix="*#39" usage_samples=";NUMBER;5000" description="Use 'Auto redial
with callback'"/>

```

```

    <feature_code ss_name="rbp" feature_key="test" feature="test" prefix="*#34"
default_prefix="*#34" usage_samples="PIN;1234" description="Check 'Restricting call-processes
by NI and PIN'"/>
    <feature_code ss_name="clip" feature_key="test" feature="test" prefix="*#30"
default_prefix="*#30" usage_samples="" description="Check 'Calling Line Identification
Presentation'"/>
    <feature_code ss_name="clir" feature_key="test" feature="test" prefix="*#31"
default_prefix="*#31" usage_samples="" description="Check 'Calling Line Identification
Restriction'"/>
    <feature_code ss_name="message" feature_key="manage" feature="use" prefix="*#75"
default_prefix="*#75" usage_samples="NUMBER;ACTION;5000;1" description="Manage by received
messages"/>
    <feature_code ss_name="redial" feature_key="test" feature="test" prefix="*#77"
default_prefix="*#77" usage_samples="" description="Check 'Last number redial'"/>
    <feature_code ss_name="voice_page_control" feature_key="test" feature="test" prefix="*
#82" default_prefix="*#82" usage_samples="" description="Check 'Voice Page Control' service"/>
    <feature_code ss_name="speed_dial" feature_key="test" feature="test" prefix="*#51"
default_prefix="*#51" usage_samples="CODE;CODE NUMBER;1;1 5000" description="Check 'Speed dial
(Abbreviated dialing)"/>
    <feature_code ss_name="direct_call" feature_key="test" feature="test" prefix="*#53"
default_prefix="*#53" usage_samples="NUMBER;5000" description="Check 'Direct call'"/>
    <feature_code ss_name="alarm" feature_key="test" feature="test" prefix="*#55"
default_prefix="*#55" usage_samples="ARM_NUMBER" description="Check alarm status"/>
    <feature_code ss_name="boss_group" feature_key="test" feature="test" prefix="*#521"
default_prefix="*#521" usage_samples="BOSS_GROUP;BOSS_GROUP NUMBER;4000;4000 5000" description="
Check 'Call Forwarding Unconditional' for 'Boss-group'"/>
    <feature_code ss_name="boss_group" feature_key="test" feature="test" prefix="*#522"
default_prefix="*#521" usage_samples="BOSS_GROUP;BOSS_GROUP NUMBER;4000;4000 5000" description="
Check 'Call Forwarding Unconditional' for 'Boss-group'"/>
    <feature_code ss_name="boss_group" feature_key="test" feature="test" prefix="*#527"
default_prefix="*#521" usage_samples="BOSS_GROUP;BOSS_GROUP NUMBER;4000;4000 5000" description="
Check 'Call Forwarding Unconditional' for 'Boss-group'"/>
    <feature_code ss_name="boss_group" feature_key="test" feature="test" prefix="*#528"
default_prefix="*#521" usage_samples="BOSS_GROUP;BOSS_GROUP NUMBER;4000;4000 5000" description="
Check 'Call Forwarding Unconditional' for 'Boss-group'"/>
    <feature_code ss_name="rfc" feature_key="test" feature="test" prefix="*#64"
default_prefix="*#64" usage_samples="" description="Check 'Rejection of Forwarded Calls'"/>
    <feature_code ss_name="chold" feature_key="test" feature="test" prefix="*#94"
default_prefix="*#94" usage_samples="" description="Check 'Call hold'"/>
    <feature_code ss_name="ctr" feature_key="test" feature="test" prefix="*#96"
default_prefix="*#96" usage_samples="" description="Check 'Call Transfer'"/>
    <feature_code ss_name="3way" feature_key="test" feature="test" prefix="*#95"
default_prefix="*#95" usage_samples="" description="Check '3-Way Conference'"/>
    <feature_code ss_name="cfu_type2" feature_key="activate" feature="activate" prefix="*4
1" default_prefix="*41" usage_samples="NUMBER;MODE NUMBER;5000;1 5000" description="Activate
'Call Forwarding Unconditional without notification'"/>
    <feature_code ss_name="cfnr_type2" feature_key="activate" feature="activate" prefix="*
42" default_prefix="*42" usage_samples="NUMBER;MODE NUMBER;5000;1 5000" description="Activate
'Call Forwarding No Reply without notification'"/>

```

```

    <feature_code ss_name="callback" feature_key="use" feature="use" prefix="*40"
default_prefix="*40" usage_samples="" description="Use 'CallBack'"/>
    <feature_code ss_name="cw" feature_key="activate" feature="activate" prefix="*43"
default_prefix="*43" usage_samples="" description="Activate 'Call Waiting'"/>
    <feature_code ss_name="teleconference" feature_key="create_room" feature="teleconferen
ce" prefix="*470" default_prefix="*470" usage_samples="ROOM" description="Create the
teleconference room"/>
    <feature_code ss_name="teleconference" feature_key="destroy_room" feature="teleconfere
nce" prefix="*471" default_prefix="*471" usage_samples="ROOM" description="Destroy the
teleconference room"/>
    <feature_code ss_name="teleconference" feature_key="enter_room" feature="teleconferenc
e" prefix="*472" default_prefix="*472" usage_samples="ROOM" description="Enter the
teleconference room"/>
    <feature_code ss_name="teleconference" feature_key="call_subscriber" feature="teleconf
erence" prefix="*473" default_prefix="*473" usage_samples="ROOM SUBSCRIBER" description="Call
subscriber to the teleconference room"/>
    <feature_code ss_name="teleconference" feature_key="cancel_subscriber" feature="teleco
nference" prefix="*474" default_prefix="*474" usage_samples="ROOM SUBSCRIBER" description="Can
cel subscriber from the teleconference room"/>
    <feature_code ss_name="teleconference" feature_key="change_subscriber_voice" feature="
teleconference" prefix="*475" default_prefix="*475" usage_samples="ROOM SUBSCRIBER"
description="Change subscriber voice(trigger mode) of the teleconference room"/>
    <feature_code ss_name="teleconference" feature_key="activate_subscriber_voice" feature="
teleconference" prefix="*476" default_prefix="*476" usage_samples="ROOM SUBSCRIBER"
description="Activate subscriber voice of the teleconference room"/>
    <feature_code ss_name="teleconference" feature_key="deactivate_subscriber_voice"
feature="teleconference" prefix="*477" default_prefix="*477" usage_samples="ROOM SUBSCRIBER"
description="Deactivate subscriber voice of the teleconference room"/>
    <feature_code ss_name="teleconference" feature_key="call_group" feature="teleconferenc
e" prefix="*478" default_prefix="*478" usage_samples="ROOM GROUP" description="Call group of
subscribers to the teleconference room"/>
    <feature_code ss_name="teleconference" feature_key="cancel_group" feature="teleconfere
nce" prefix="*479" default_prefix="*479" usage_samples="ROOM GROUP" description="Cancel group
of subscribers from the teleconference room"/>
    <feature_code ss_name="teleconference" feature_key="create_one_of_rooms" feature="tele
conference" prefix="*480" default_prefix="*480" usage_samples="ROOM" description="Create one
of the teleconference rooms"/>
    <feature_code ss_name="follow_me_no_response" feature_key="set_timeout" feature="activ
ate" prefix="*125" default_prefix="*125" usage_samples="TIMEOUT;15" description="Set No
Response timeout for 'Follow me No Response'"/>
    <feature_code ss_name="my_number" feature_key="use" feature="use" prefix="*123"
default_prefix="*123" usage_samples="" description="Use 'Get number of my phone'"/>
    <feature_code ss_name="boss_group" feature_key="use_line" feature="use" prefix="*126"
default_prefix="*126" usage_samples="BOSS_GROUP;BOSS_GROUP LINE;4000;4000 1" description="Use
'Boss-group call'"/>
    <feature_code ss_name="cc_agent" feature_key="login" feature="use" prefix="*160"
default_prefix="*160" usage_samples="AGENT_ID PASSWORD;10 1234" description="Login Contact-
Center Agent"/>

```



```

    <feature_code ss_name="cc_agent" feature_key="supervise" feature="use" prefix="*164"
default_prefix="*164" usage_samples="AGENT_ID MODE;10;1" description="Supervise Contact-Center
Agent"/>
    <feature_code ss_name="cc_agent" feature_key="call_agent" feature="use" prefix="*165"
default_prefix="*165" usage_samples="AGENT_ID" description="Call Contact-Center Agent"/>
    <feature_code ss_name="cc_agent" feature_key="supervise2" feature="use" prefix="*166"
default_prefix="*166" usage_samples="AGENT_ID MODE;10;1" description="Supervise Contact-Center
Agent"/>
    <feature_code ss_name="cc_agent" feature_key="set_default_supervise_mode" feature="use
" prefix="*167" default_prefix="*167" usage_samples="MODE" description="Set default supervise
mode of Contact-Center supervisor"/>
    <feature_code ss_name="acb" feature_key="activate" feature="activate" prefix="*16"
default_prefix="*16" usage_samples="" description="Activate 'Anonymous Call Blocking'"/>
    <feature_code ss_name="time_service" feature_key="use" feature="use" prefix="*100"
default_prefix="*100" usage_samples="" description="Use 'Get current time'"/>
    <feature_code ss_name="teleconference_manager" feature_key="member" feature="teleconfe
rence" prefix="*171" default_prefix="*171" usage_samples="MEETING_ID;1234" description="Join
to the teleconference meeting by member"/>
    <feature_code ss_name="flexicall" feature_key="activate" feature="activate" prefix="*1
77" default_prefix="*177" usage_samples=";CODE NUMBER;1 5000" description="Activate
'FlexiCall'"/>
    <feature_code ss_name="privacy" feature_key="activate" feature="activate" prefix="*186
" default_prefix="*186" usage_samples="" description="Activate always_on 'Privacy' mode"/>
    <feature_code ss_name="mcid" feature_key="use" feature="use" prefix="*36"
default_prefix="*36" usage_samples="" description="Use 'Malicious Call Identification'"/>
    <feature_code ss_name="auto_redial" feature_key="use_busy" feature="use" prefix="*37"
default_prefix="*37" usage_samples="NUMBER;5000" description="Use 'Auto redial' in busy mode"/
>
    <feature_code ss_name="auto_redial" feature_key="use_all" feature="use" prefix="*38"
default_prefix="*38" usage_samples="NUMBER;5000" description="Use 'Auto redial' in any mode"/>
    <feature_code ss_name="auto_redial_with_callback" feature_key="start" feature="use"
prefix="*39" default_prefix="*39" usage_samples=";NUMBER;5000" description="Use 'Auto redial
with callback'"/>
    <feature_code ss_name="rbp" feature_key="activate" feature="activate" prefix="*34"
default_prefix="*34" usage_samples="PIN CODE;1234 01" description="Activate 'Restricting call-
processes by NI and PIN'"/>
    <feature_code ss_name="rbp" feature_key="use" feature="use" prefix="*32"
default_prefix="*32" usage_samples="PIN NUMBER;PIN;1234 5000;5000" description="Use
'Restricting call-processes by NI and PIN'"/>
    <feature_code ss_name="clir" feature_key="activate" feature="activate" prefix="*31"
default_prefix="*31" usage_samples="" description="Activate 'Calling Line Identification
Restriction'"/>
    <feature_code ss_name="message" feature_key="allocate" feature="use" prefix="*75"
default_prefix="*75" usage_samples="NUMBER;MESSAGE_NUMBER;5000;1" description="Send message"/>
    <feature_code ss_name="redial" feature_key="activate" feature="activate" prefix="*77"
default_prefix="*77" usage_samples="" description="Activate 'Last number redial'"/>
    <feature_code ss_name="conference" feature_key="use" feature="teleconference" prefix="
*71" default_prefix="*71" usage_samples="" description="Use 'Conference Call, Add-on'"/>
    <feature_code ss_name="fax_to_email" feature_key="use" feature="ivr" prefix="*73"
default_prefix="*73" usage_samples="" description="Fax to email service"/>

```

```

    <feature_code ss_name="remote_phone" feature_key="activate" feature="activate" prefix="
*70" default_prefix="*70" usage_samples="PIN LOCAL_NUMBER;1234 5000" description="Activate
'Remote phone'"/>
    <feature_code ss_name="voice_page" feature_key="use" feature="use" prefix="*80"
default_prefix="*80" usage_samples=";NUMBER;5000" description="Use 'VoicePage' service"/>
    <feature_code ss_name="voice_page_control" feature_key="activate" feature="activate"
prefix="*82" default_prefix="*82" usage_samples="" description="Activate 'Voice Page Control'
service"/>
    <feature_code ss_name="walkie_talkie" feature_key="use" feature="use" prefix="*83"
default_prefix="*83" usage_samples="MODE NUMBER;NUMBER;1 5000;5000" description="Use 'Walkie
Talkie' service"/>
    <feature_code ss_name="intercom" feature_key="use" feature="use" prefix="*85"
default_prefix="*85" usage_samples="NUMBER;NUMBER;5000" description="Use 'Intercom' service"/>
    <feature_code ss_name="zone_page" feature_key="use" feature="use" prefix="*81"
default_prefix="*81" usage_samples=";NUMBER;5000" description="Use 'ZonePage' service in
Conference mode"/>
    <feature_code ss_name="speed_dial" feature_key="activate" feature="activate" prefix="*
51" default_prefix="*51" usage_samples="CODE NUMBER;1 5000" description="Activate 'Speed dial
(Abbreviated dialing)'" />
    <feature_code ss_name="direct_call" feature_key="activate" feature="activate" prefix="
*53" default_prefix="*53" usage_samples="NUMBER;5000" description="Activate 'Direct call'"/>
    <feature_code ss_name="alarm" feature_key="activate_one_time" feature="activate"
prefix="*55" default_prefix="*55" usage_samples="ARM_NUMBER DAY HHMM;ARM_NUMBER HHMM;1 5
0900;1 0900" description="Activate one-time alarm"/>
    <feature_code ss_name="alarm" feature_key="activate_daily" feature="activate" prefix="
*56" default_prefix="*56" usage_samples="ARM_NUMBER DAYS HHMM;ARM_NUMBER HHMM;1 12345 0900;1
0900" description="Activate daily alarm"/>
    <feature_code ss_name="park" feature_key="use_park" feature="ivr" prefix="*57"
default_prefix="*57" usage_samples=";SLOT_NUMBER;15" description="Put a call on park"/>
    <feature_code ss_name="park" feature_key="use_unpark" feature="ivr" prefix="*58"
default_prefix="*58" usage_samples=";SLOT_NUMBER;15" description="Pick up the parked call"/>
    <feature_code ss_name="boss_group" feature_key="activate" feature="activate" prefix="*
521" default_prefix="*521" usage_samples="BOSS_GROUP NUMBER;4000 5000" description="Activate
'Call Forwarding Unconditional' for 'Boss-group'"/>
    <feature_code ss_name="boss_group" feature_key="activate" feature="activate" prefix="*
522" default_prefix="*521" usage_samples="BOSS_GROUP NUMBER;4000 5000" description="Activate
'Call Forwarding Unconditional' for 'Boss-group'"/>
    <feature_code ss_name="boss_group" feature_key="activate" feature="activate" prefix="*
527" default_prefix="*521" usage_samples="BOSS_GROUP NUMBER;4000 5000" description="Activate
'Call Forwarding Unconditional' for 'Boss-group'"/>
    <feature_code ss_name="boss_group" feature_key="activate" feature="activate" prefix="*
528" default_prefix="*521" usage_samples="BOSS_GROUP NUMBER;4000 5000" description="Activate
'Call Forwarding Unconditional' for 'Boss-group'"/>
    <feature_code ss_name="speed_dial" feature_key="use" feature="use" prefix="*"
default_prefix="*" usage_samples="CODE;1" description="Use 'Speed dial (Abbreviated
dialing)'" />
    <feature_code ss_name="sca" feature_key="activate" feature="activate" prefix="*60"
default_prefix="*60" usage_samples="PIN;PIN NUMBER;PIN INDEX NUMBER" description="Activate
'Selective Call Acceptance'"/>

```

```

    <feature_code ss_name="scr" feature_key="activate" feature="activate" prefix="*61"
default_prefix="*61" usage_samples="PIN;PIN NUMBER;PIN INDEX NUMBER" description="Activate
'Selective Call Rejection'"/>
    <feature_code ss_name="sco_white" feature_key="activate" feature="activate" prefix="*6
2" default_prefix="*62" usage_samples="PIN;PIN NUMBER;PIN INDEX NUMBER" description="Activate
'Selective Call Originating, White list'"/>
    <feature_code ss_name="sco_black" feature_key="activate" feature="activate" prefix="*6
3" default_prefix="*63" usage_samples="PIN;PIN NUMBER;PIN INDEX NUMBER" description="Activate
'Selective Call Originating, Black list'"/>
    <feature_code ss_name="rfc" feature_key="activate" feature="activate" prefix="*64"
default_prefix="*64" usage_samples="" description="Activate 'Rejection of Forwarded Calls'"/>
    <feature_code ss_name="chold" feature_key="activate" feature="activate" prefix="*94"
default_prefix="*94" usage_samples="" description="Activate 'Call hold'"/>
    <feature_code ss_name="ctr" feature_key="activate" feature="activate" prefix="*96"
default_prefix="*96" usage_samples="" description="Activate 'Call Transfer'"/>
    <feature_code ss_name="3way" feature_key="activate" feature="activate" prefix="*95"
default_prefix="*95" usage_samples="" description="Activate '3-Way Conference'"/>
    <feature_code ss_name="voicemail" feature_key="use_reader" feature="use" prefix="*90"
default_prefix="*90" usage_samples="" description="Check and play voice mail"/>
    <feature_code ss_name="voicemail" feature_key="use_reader_other_number" feature="use"
prefix="*91" default_prefix="*91" usage_samples=";NUMBER" description="Check and play voice
mail from any number"/>
    <feature_code ss_name="vip_call" feature_key="use" feature="use" prefix="*99"
default_prefix="*99" usage_samples="" description="Use VIP-call service"/>
    <feature_code ss_name="pickup" feature_key="use_any" feature="use" prefix="*08"
default_prefix="*08" usage_samples="" description="Use 'Pickup' any number in group(s)"/>
    <feature_code ss_name="pickup" feature_key="use_direct" feature="use" prefix="*07"
default_prefix="*07" usage_samples="NUMBER;GROUP_ID;5000;5" description="Use 'Pickup' direct
number or group"/>
    <feature_code ss_name="intervention" feature_key="use" feature="use" prefix="*09"
default_prefix="*09" usage_samples="MODE NUMBER;NUMBER;0 5000;5000" description="Use
'Intervention'"/>
    <feature_code ss_name="second_handset" feature_key="use" feature="second_handset"
prefix="*0" default_prefix="*0" usage_samples="" description="Use 'Second Handset Call'"/>
    <feature_code ss_name="redial" feature_key="use" feature="use" prefix=""
default_prefix="" usage_samples="" description="Use 'Last number redial'"/>
    <feature_code ss_name="cf_aon" feature_key="deactivate" feature="deactivate" prefix="#
20" default_prefix="#20" usage_samples="" description="Deactivate 'Call Forwarding AON'"/>
    <feature_code ss_name="cfu" feature_key="deactivate" feature="deactivate" prefix="#21"
default_prefix="#21" usage_samples="" description="Deactivate 'Call Forwarding
Unconditional'"/>
    <feature_code ss_name="cfb" feature_key="deactivate" feature="deactivate" prefix="#22"
default_prefix="#22" usage_samples="" description="Deactivate 'Call Forwarding Busy'"/>
    <feature_code ss_name="cfos" feature_key="deactivate" feature="deactivate" prefix="#24
" default_prefix="#24" usage_samples="" description="Deactivate 'Call Forwarding Out of
Service'"/>
    <feature_code ss_name="cfnr" feature_key="deactivate" feature="deactivate" prefix="#27
" default_prefix="#27" usage_samples="" description="Deactivate 'Call Forwarding No Reply'"/>

```



```

    <feature_code ss_name="cft" feature_key="deactivate" feature="deactivate" prefix="#28"
    default_prefix="#28" usage_samples=";DAYS;12345" description="Deactivate 'Call Forwarding
Unconditional Time Dependent'"/>
    <feature_code ss_name="follow_me" feature_key="deactivate" feature="deactivate" prefix="
#23" default_prefix="#23" usage_samples=";PIN; LOCAL_NUMBER;PIN LOCAL_NUMBER;PIN MODE
LOCAL_NUMBER;1234; 5000;1234 5000;1234 1 5000" description="Deactivate 'Follow me'"/>
    <feature_code ss_name="follow_me_no_response" feature_key="deactivate" feature="deacti
vate" prefix="#25" default_prefix="#25" usage_samples=";PIN; LOCAL_NUMBER;PIN LOCAL_NUMBER;PIN
MODE LOCAL_NUMBER;1234; 5000;1234 5000;1234 1 5000" description="Deactivate 'Follow me No
Response'"/>
    <feature_code ss_name="dnd" feature_key="deactivate" feature="deactivate" prefix="#26"
    default_prefix="#26" usage_samples="" description="Deactivate 'Do not Disturb'"/>
    <feature_code ss_name="cfu_type2" feature_key="deactivate" feature="deactivate" prefix="
#41" default_prefix="#41" usage_samples="NUMBER;5000" description="Deactivate 'Call Forwarding
Unconditional without notification'"/>
    <feature_code ss_name="cfnr_type2" feature_key="deactivate" feature="deactivate"
prefix="#42" default_prefix="#42" usage_samples="NUMBER;5000" description="Deactivate 'Call
Forwarding No Reply without notification'"/>
    <feature_code ss_name="cw" feature_key="deactivate" feature="deactivate" prefix="#43"
default_prefix="#43" usage_samples="" description="Deactivate 'Call Waiting'"/>
    <feature_code ss_name="message" feature_key="free" feature="use" prefix="#75"
default_prefix="#75" usage_samples="NUMBER;MESSAGE_NUMBER;5000;1" description="Remove received
message"/>
    <feature_code ss_name="redial" feature_key="deactivate" feature="deactivate" prefix="#
77" default_prefix="#77" usage_samples="" description="Deactivate 'Last number redial'"/>
    <feature_code ss_name="remote_phone" feature_key="deactivate" feature="deactivate"
prefix="#70" default_prefix="#70" usage_samples="PIN;1234" description="Deactivate 'Remote
phone'"/>
    <feature_code ss_name="voice_page_control" feature_key="deactivate" feature="deactivat
e" prefix="#82" default_prefix="#82" usage_samples="" description="Deactivate 'Voice Page
Control' service"/>
    <feature_code ss_name="speed_dial" feature_key="deactivate" feature="deactivate"
prefix="#51" default_prefix="#51" usage_samples="CODE;1" description="Deactivate 'Speed dial
(Abbreviated dialing)"/>
    <feature_code ss_name="direct_call" feature_key="deactivate" feature="deactivate"
prefix="#53" default_prefix="#53" usage_samples="" description="Deactivate 'Direct call'"/>
    <feature_code ss_name="alarm" feature_key="deactivate" feature="deactivate" prefix="#5
5" default_prefix="#55" usage_samples="ARM_NUMBER;0;1" description="Deactive alarm (one or
all)"/>
    <feature_code ss_name="boss_group" feature_key="deactivate" feature="deactivate"
prefix="#521" default_prefix="#521" usage_samples="BOSS_GROUP;4000" description="Deactivate
'Call Forwarding Unconditional' for 'Boss-group'"/>
    <feature_code ss_name="boss_group" feature_key="deactivate" feature="deactivate"
prefix="#522" default_prefix="#521" usage_samples="BOSS_GROUP;4000" description="Deactivate
'Call Forwarding Unconditional' for 'Boss-group'"/>
    <feature_code ss_name="boss_group" feature_key="deactivate" feature="deactivate"
prefix="#527" default_prefix="#521" usage_samples="BOSS_GROUP;4000" description="Deactivate
'Call Forwarding Unconditional' for 'Boss-group'"/>

```

```

    <feature_code ss_name="boss_group" feature_key="deactivate" feature="deactivate"
prefix="#528" default_prefix="#521" usage_samples="BOSS_GROUP;4000" description="Deactivate
'Call Forwarding Unconditional' for 'Boss-group'"/>
    <feature_code ss_name="ALL" feature_key="deactivate" feature="deactivate" prefix="#50"
default_prefix="#50" usage_samples="PIN" description="Deactivate all supplementary services"/
>

    <feature_code ss_name="sca" feature_key="deactivate" feature="deactivate" prefix="#60"
default_prefix="#60" usage_samples="PIN;PIN 0;PIN NUMBER;PIN INDEX" description="Deactivate
'Selective Call Acceptance'"/>
    <feature_code ss_name="scr" feature_key="deactivate" feature="deactivate" prefix="#61"
default_prefix="#61" usage_samples="PIN;PIN 0;PIN NUMBER;PIN INDEX" description="Deactivate
'Selective Call Rejection'"/>
    <feature_code ss_name="sco_white" feature_key="deactivate" feature="deactivate" prefix="
#62" default_prefix="#62" usage_samples="PIN;PIN 0;PIN NUMBER;PIN INDEX" description="Deactiva
te 'Selective Call Originating, White list'"/>
    <feature_code ss_name="sco_black" feature_key="deactivate" feature="deactivate" prefix="
#63" default_prefix="#63" usage_samples="PIN;PIN 0;PIN NUMBER;PIN INDEX" description="Deactiva
te 'Selective Call Originating, Black list'"/>
    <feature_code ss_name="rfc" feature_key="deactivate" feature="deactivate" prefix="#64"
default_prefix="#64" usage_samples="" description="Deactivate 'Rejection of Forwarded Calls'
"/>

    <feature_code ss_name="rbp" feature_key="deactivate" feature="deactivate" prefix="#34"
default_prefix="#34" usage_samples="PIN;1234" description="Deactivate 'Restricting call-
processes by NI and PIN'"/>
    <feature_code ss_name="clir" feature_key="deactivate" feature="deactivate" prefix="#31
" default_prefix="#31" usage_samples="" description="Deactivate 'Calling Line Identification
Restriction'"/>
    <feature_code ss_name="cc_agent" feature_key="logout" feature="use" prefix="#160"
default_prefix="#160" usage_samples="" description="Logout Contact-Center Agent"/>
    <feature_code ss_name="cc_agent" feature_key="complete" feature="use" prefix="#161"
default_prefix="#161" usage_samples="" description="Complete conversation Contact-Center
Agent"/>
    <feature_code ss_name="cc_agent" feature_key="enter_auxwork" feature="use" prefix="#16
2" default_prefix="#162" usage_samples="" description="Enter AuxWork Contact-Center Agent"/>
    <feature_code ss_name="cc_agent" feature_key="make_available" feature="use" prefix="#1
63" default_prefix="#163" usage_samples="" description="Make Available Contact-Center Agent"/>
    <feature_code ss_name="acb" feature_key="deactivate" feature="deactivate" prefix="#16"
default_prefix="#16" usage_samples="" description="Deactivate 'Anonymous Call Blocking'"/>
    <feature_code ss_name="flexicall" feature_key="deactivate" feature="deactivate" prefix="
#177" default_prefix="#177" usage_samples="" description="Deactivate 'FlexiCall'"/>
    <feature_code ss_name="chold" feature_key="deactivate" feature="deactivate" prefix="#9
4" default_prefix="#94" usage_samples="" description="Deactivate 'Call hold'"/>
    <feature_code ss_name="ctr" feature_key="deactivate" feature="deactivate" prefix="#96"
default_prefix="#96" usage_samples="" description="Deactivate 'Call Transfer'"/>
    <feature_code ss_name="3way" feature_key="deactivate" feature="deactivate" prefix="#95
" default_prefix="#95" usage_samples="" description="Deactivate '3-Way Conference'"/>
    <feature_code ss_name="cfu" feature_key="activate_pulse" feature="activate" prefix="11
21" default_prefix="1121" usage_samples="" description="Activate 'Call Forwarding
Unconditional' (pulse)"/>

```

```

    <feature_code ss_name="cfb" feature_key="activate_pulse" feature="activate" prefix="11
22" default_prefix="1122" usage_samples="" description="Activate 'Call Forwarding
Busy' (pulse)"/>
    <feature_code ss_name="cfos" feature_key="activate_pulse" feature="activate" prefix="1
124" default_prefix="1124" usage_samples="" description="Activate 'Call Forwarding Out of
Service' (pulse)"/>
    <feature_code ss_name="cfnr" feature_key="activate_pulse" feature="activate" prefix="1
127" default_prefix="1127" usage_samples="" description="Activate 'Call Forwarding No
Reply' (pulse)"/>
    <feature_code ss_name="dnd" feature_key="activate_pulse" feature="activate" prefix="11
26" default_prefix="1126" usage_samples="" description="Activate 'Do not Disturb' (pulse)"/>
    <feature_code ss_name="rfc" feature_key="activate_pulse" feature="activate" prefix="11
64" default_prefix="1164" usage_samples="" description="Activate 'Rejection of Forwarded
Calls' (pulse)"/>
    <feature_code ss_name="acb" feature_key="activate_pulse" feature="activate" prefix="11
16" default_prefix="1116" usage_samples="" description="Activate 'Anonymous Call
Blocking' (pulse)"/>
    <feature_code ss_name="chold" feature_key="activate_pulse" feature="activate" prefix="
1194" default_prefix="1194" usage_samples="" description="Activate 'Call hold' (pulse)"/>
    <feature_code ss_name="ctr" feature_key="activate_pulse" feature="activate" prefix="11
96" default_prefix="1196" usage_samples="" description="Activate 'Call Transfer' (pulse)"/>
    <feature_code ss_name="3way" feature_key="activate_pulse" feature="activate" prefix="1
195" default_prefix="1195" usage_samples="" description="Activate '3-Way Conference' (pulse)"/
>
    <feature_code ss_name="cw" feature_key="activate_pulse" feature="activate" prefix="114
3" default_prefix="1143" usage_samples="" description="Activate 'Call Waiting' (pulse)"/>
    <feature_code ss_name="clir" feature_key="activate_pulse" feature="activate" prefix="1
131" default_prefix="1131" usage_samples="" description="Activate 'Calling Line Identification
Restriction' (pulse)"/>
    <feature_code ss_name="cfu" feature_key="deactivate_pulse" feature="deactivate" prefix="
1221" default_prefix="1221" usage_samples="" description="Deactivate 'Call Forwarding
Unconditional' (pulse)"/>
    <feature_code ss_name="cfb" feature_key="deactivate_pulse" feature="deactivate" prefix="
1222" default_prefix="1222" usage_samples="" description="Deactivate 'Call Forwarding
Busy' (pulse)"/>
    <feature_code ss_name="cfos" feature_key="deactivate_pulse" feature="deactivate"
prefix="1224" default_prefix="1224" usage_samples="" description="Deactivate 'Call Forwarding
Out of Service' (pulse)"/>
    <feature_code ss_name="cfnr" feature_key="deactivate_pulse" feature="deactivate"
prefix="1227" default_prefix="1227" usage_samples="" description="Deactivate 'Call Forwarding
No Reply' (pulse)"/>
    <feature_code ss_name="dnd" feature_key="deactivate_pulse" feature="deactivate" prefix="
1226" default_prefix="1226" usage_samples="" description="Deactivate 'Do not Disturb' (pulse)
"/>
    <feature_code ss_name="rfc" feature_key="deactivate_pulse" feature="deactivate" prefix="
1264" default_prefix="1264" usage_samples="" description="Deactivate 'Rejection of Forwarded
Calls' (pulse)"/>
    <feature_code ss_name="acb" feature_key="deactivate_pulse" feature="deactivate" prefix="
1216" default_prefix="1216" usage_samples="" description="Deactivate 'Anonymous Call
Blocking' (pulse)"/>

```

```

    <feature_code ss_name="chold" feature_key="deactivate_pulse" feature="deactivate"
prefix="1294" default_prefix="1294" usage_samples="" description="Deactivate 'Call
hold' (pulse)"/>
    <feature_code ss_name="ctr" feature_key="deactivate_pulse" feature="deactivate" prefix="
1296" default_prefix="1296" usage_samples="" description="Deactivate 'Call Transfer' (pulse)"/
>
    <feature_code ss_name="3way" feature_key="deactivate_pulse" feature="deactivate"
prefix="1295" default_prefix="1295" usage_samples="" description="Deactivate '3-Way
Conference' (pulse)"/>
    <feature_code ss_name="cw" feature_key="deactivate_pulse" feature="deactivate" prefix="
1243" default_prefix="1243" usage_samples="" description="Deactivate 'Call Waiting' (pulse)"/>
    <feature_code ss_name="clir" feature_key="deactivate_pulse" feature="deactivate"
prefix="1231" default_prefix="1231" usage_samples="" description="Deactivate 'Calling Line
Identification Restriction' (pulse)"/>
    <feature_code ss_name="cfu" feature_key="test_pulse" feature="test" prefix="1321"
default_prefix="1321" usage_samples="" description="Check 'Call Forwarding Unconditional' in
common (pulse)"/>
    <feature_code ss_name="cfb" feature_key="test_pulse" feature="test" prefix="1322"
default_prefix="1322" usage_samples="" description="Check 'Call Forwarding Busy' (pulse)"/>
    <feature_code ss_name="cfos" feature_key="test_pulse" feature="test" prefix="1324"
default_prefix="1324" usage_samples="" description="Check 'Call Forwarding Out of
Service' (pulse)"/>
    <feature_code ss_name="cfnr" feature_key="test_pulse" feature="test" prefix="1327"
default_prefix="1327" usage_samples="" description="Check 'Call Forwarding No Reply' (pulse)"/
>
    <feature_code ss_name="dnd" feature_key="test_pulse" feature="test" prefix="1326"
default_prefix="1326" usage_samples="" description="Check 'Do not Disturb' (pulse)"/>
    <feature_code ss_name="rfc" feature_key="test_pulse" feature="test" prefix="1364"
default_prefix="1364" usage_samples="" description="Check 'Rejection of Forwarded
Calls' (pulse)"/>
    <feature_code ss_name="acb" feature_key="test_pulse" feature="test" prefix="1316"
default_prefix="1316" usage_samples="" description="Check 'Anonymous Call Blocking' (pulse)"/>
    <feature_code ss_name="chold" feature_key="test_pulse" feature="test" prefix="1394"
default_prefix="1394" usage_samples="" description="Check 'Call hold' (pulse)"/>
    <feature_code ss_name="ctr" feature_key="test_pulse" feature="test" prefix="1396"
default_prefix="1396" usage_samples="" description="Check 'Call Transfer' (pulse)"/>
    <feature_code ss_name="3way" feature_key="test_pulse" feature="test" prefix="1395"
default_prefix="1395" usage_samples="" description="Check '3-Way Conference' (pulse)"/>
    <feature_code ss_name="cw" feature_key="test_pulse" feature="test" prefix="1343"
default_prefix="1343" usage_samples="" description="Check 'Call Waiting' (pulse)"/>
    <feature_code ss_name="clir" feature_key="test_pulse" feature="test" prefix="1331"
default_prefix="1331" usage_samples="" description="Check 'Calling Line Identification
Restriction' (pulse)"/>
    <feature_code ss_name="cfu" feature_key="test_number_pulse" feature="test" prefix="142
1" default_prefix="1421" usage_samples="" description="Check 'Call Forwarding Unconditional'
on specific number (pulse)"/>
    <feature_code ss_name="cfb" feature_key="test_number_pulse" feature="test" prefix="142
2" default_prefix="1422" usage_samples="" description="Check 'Call Forwarding Busy' on
specific number (pulse)"/>

```

```

    <feature_code ss_name="cfos" feature_key="test_number_pulse" feature="test" prefix="14
24" default_prefix="1424" usage_samples="" description="Check 'Call Forwarding Out of Service'
on specific number (pulse)"/>
    <feature_code ss_name="cfnr" feature_key="test_number_pulse" feature="test" prefix="14
27" default_prefix="1427" usage_samples="" description="Check 'Call Forwarding No Reply' on
specific number (pulse)"/>
    <feature_code ss_name="ALL" feature_key="deactivate_pulse" feature="deactivate" prefix="
1450" default_prefix="1450" usage_samples="" description="Deactivate all supplementary
services (pulse)"/>
  </result>
</out>

```

Нс_ss_feature_code_set

Команда позволяет изменить Feature коды для работы с услугами в рамках определенного домена.

Аналог команды в CoCon:

domain/**<DOMAIN>**/**ss/feature-codes/set**

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_feature_code_set

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set supplementary services' feature codes for specific domain.
      Usage: http://server:9999/commands/ss\_feature\_code\_set
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <!-- non-empty token -->
  <xs:simpleType name="prefixType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
  <!-- featureCodeType -->
  <xs:complexType name="featureCodeType">
    <xs:attribute name="ss_name" type="xs:string" use="required"/>
    <xs:attribute name="feature_key" type="xs:string" use="required"/>
    <xs:attribute name="feature" type="xs:string" use="optional"/>
    <xs:attribute name="prefix" type="prefixType" use="required"/>
    <xs:attribute name="default_prefix" type="prefixType" use="optional"/>
    <xs:attribute name="usage_samples" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- okType -->
  <xs:complexType name="okType"/>
  <!-- duplicatesType -->
  <xs:complexType name="duplicatesType">
    <xs:sequence>
      <xs:element name="feature_code" type="featureCodeType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="feature_code" type="featureCodeType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="result">
        <xs:complexType>
          <xs:choice>
            <xs:element name="ok" type="okType"/>
            <xs:element name="duplicates" type="duplicatesType"/>
          </xs:choice>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1

Запрос:

http://192.168.1.21:9999/commands/ss_feature_code_set

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local">
    <feature_code ss_name="cfu" feature_key="activate_pulse" prefix="*111"/>
    <feature_code ss_name="cfu" feature_key="deactivate_pulse" prefix="#222"/>
    <feature_code ss_name="cfb" feature_key="test_number_pulse" prefix="#333"/>
  </request>
</in>

```

Ответ:200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_feature_code_set.xsd">
  <result>
    <ok/>
  </result>
</out>

```

Пример 2

Запрос:

http://192.168.1.21:9999/commands/ss_feature_code_set


```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_feature_code_set.xsd">
  <result>
    <duplicates>
      <feature_code ss_name="cfb" feature_key="activate_pulse" feature="activate"
        prefix="*111" default_prefix="*111" usage_samples="" description="Activate 'Call Forwarding
        Busy' (pulse)"/>
      <feature_code ss_name="cfb" feature_key="deactivate_pulse" feature="deactivate"
        prefix="#222" default_prefix="#222" usage_samples="" description="Deactivate 'Call Forwarding
        Busy' (pulse)"/>
      <feature_code ss_name="cfb" feature_key="test_number_pulse" feature="test"
        prefix="#333" default_prefix="#333" usage_samples="" description="Check 'Call Forwarding Busy' on
        specific number (pulse)"/>
      <feature_code ss_name="cfnr" feature_key="test_number_pulse" feature="test"
        prefix="#333" default_prefix="#333" usage_samples="" description="Check 'Call Forwarding No
        Reply' on specific number (pulse)"/>
      <feature_code ss_name="cfu" feature_key="activate_pulse" feature="activate"
        prefix="*111" default_prefix="*111" usage_samples="" description="Activate 'Call Forwarding
        Unconditional' (pulse)"/>
      <feature_code ss_name="cfu" feature_key="deactivate_pulse" feature="deactivate"
        prefix="#222" default_prefix="#222" usage_samples="" description="Deactivate 'Call Forwarding
        Unconditional' (pulse)"/>
    </duplicates>
  </result>
</out>
```

Ответ: 200


```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_feature_code_set.xsd">
  <result>
    <duplicates>
      <feature_code ss_name="cfb" feature_key="activate_pulse" feature="activate"
        prefix="*111" default_prefix="*111" usage_samples="" description="Activate 'Call Forwarding
        Busy' (pulse)"/>
      <feature_code ss_name="cfb" feature_key="deactivate_pulse" feature="deactivate"
        prefix="#222" default_prefix="#222" usage_samples="" description="Deactivate 'Call Forwarding
        Busy' (pulse)"/>
      <feature_code ss_name="cfb" feature_key="test_number_pulse" feature="test"
        prefix="#333" default_prefix="#333" usage_samples="" description="Check 'Call Forwarding Busy' on
        specific number (pulse)"/>
      <feature_code ss_name="cfnr" feature_key="test_number_pulse" feature="test"
        prefix="#333" default_prefix="#333" usage_samples="" description="Check 'Call Forwarding No
        Reply' on specific number (pulse)"/>
      <feature_code ss_name="cfu" feature_key="activate_pulse" feature="activate"
        prefix="*111" default_prefix="*111" usage_samples="" description="Activate 'Call Forwarding
        Unconditional' (pulse)"/>
      <feature_code ss_name="cfu" feature_key="deactivate_pulse" feature="deactivate"
        prefix="#222" default_prefix="#222" usage_samples="" description="Deactivate 'Call Forwarding
        Unconditional' (pulse)"/>
    </duplicates>
  </result>
</out>

```

Нс_ss_feature_code_properties_info

Команда предназначена для просмотра служебных свойств (префикса Feature кодов услуг, разделителя параметров услуг) для работы с кодами услуг.

Аналог команды в CoCon:

domain/<DOMAIN>**/ss/feature-codes/properties/info**

Шаблон URL запроса:

http://server:9999/commands/ss_feature_code_properties_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>

  <!-- fcPropsType -->
  <xs:complexType name="fcProps">
    <xs:attribute name="delimiter" type="xs:string" use="required"/>
    <xs:attribute name="finisher" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="feature_code_properties" type="fcProps"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999//commands/ss_feature_code_properties_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_ss_feature_code_properties_info.xsd">
  <request domain="biysk.local" />
</in>

```

Ответ:200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ss_feature_code_properties_info.xsd">
  <result>
    <feature_code_properties delimiter="*" finisher="#"/>
  </result>
</out>
```

Hc_ss_incall_feature_code - Управление списками feature кодов

- [Hc_ss_incall_feature_code_clean](#)
- [Hc_ss_incall_feature_code_info](#)
- [Hc_ss_incall_feature_code_set](#)

Hc_ss_incall_feature_code_clean

Команда используется для сброса списка feature кодов для работы с incall услугами в рамках определенного домена.

Аналог команды в CoCon:

```
/domain/<DOMAIN>/ss/feature-codes/incall/clean
```

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/ss_incall_feature_code_clean

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Reset supplementary services' feature codes for specific domain.
      Usage: http://server:9999/commands/ss\_incall\_feature\_code\_clean
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <!-- non-empty token -->
  <xs:simpleType name="prefixType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
  <!-- featureCodeType -->
  <xs:complexType name="featureCodeType">
    <xs:attribute name="ss_name" type="xs:string" use="required"/>
    <xs:attribute name="feature_key" type="xs:string" use="required"/>
    <xs:attribute name="feature" type="xs:string" use="optional"/>
    <xs:attribute name="prefix" type="prefixType" use="optional"/>
    <xs:attribute name="default_prefix" type="prefixType" use="optional"/>
    <xs:attribute name="usage_samples" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- okType -->
  <xs:complexType name="okType"/>
  <!-- duplicatesType -->
  <xs:complexType name="duplicatesType">
    <xs:sequence>
      <xs:element name="feature_code" type="featureCodeType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="feature_code" type="featureCodeType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="result">
        <xs:complexType>
          <xs:choice>
            <xs:element name="ok" type="okType"/>
            <xs:element name="duplicates" type="duplicatesType"/>
          </xs:choice>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/ss_incall_feature_code_clean

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local">
    <feature_code ss_name="CallRecord" feature_key="use"/>
    <feature_code ss_name="Flash" feature_key="use"/>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_incall_feature_code_clean.xsd">
  <result>
    <ok/>
  </result>
</out>

```

Нс_ss_incall_feature_code_info

Команда предназначена для получения списка feature кодов для работы с incall услуг в рамках определенного домена.

Аналог команды в CoCon:

/domain/<DOMAIN>/ss/feature-codes/incall/info

Метод http запроса:

POST

Шаблон http запроса:

http://host:port/commands/ss_incall_feature_code_info

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get supplementary services' incall feature codes for specific domain.
      Usage: http://server:9999/commands/ss\_feature\_code\_info
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <!-- non-empty token -->
  <xs:simpleType name="prefixType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
  <!-- featureCodeType -->
  <xs:complexType name="featureCodeType">
    <xs:attribute name="ss_name" type="xs:string" use="required"/>
    <xs:attribute name="feature_key" type="xs:string" use="required"/>
    <xs:attribute name="feature" type="xs:string" use="required"/>
    <xs:attribute name="prefix" type="prefixType" use="required"/>
    <xs:attribute name="default_prefix" type="prefixType" use="required"/>
    <xs:attribute name="usage_samples" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="result">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="feature_code" type="featureCodeType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример :

Запрос:

```
http://192.168.1.21:9999/commands/ss_feature_code_info
```

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local"/>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_incall_feature_code_info.xsd">
  <result>
    <feature_code ss_name="CallRecord" feature_key="use" feature="use" prefix="*0#"
      default_prefix="*0#" usage_samples="" description="Start call record functionality."/>
    <feature_code ss_name="Flash" feature_key="use" feature="use" prefix="*#"
      default_prefix="*#" usage_samples="" description="Hold/Hole functionality."/>
    <feature_code ss_name="ExclusiveHold" feature_key="use" feature="use" prefix="*187"
      default_prefix="*187" usage_samples="" description="Boss-group's exclusive hold."/>
    <feature_code ss_name="Privacy" feature_key="activate" feature="use" prefix="*186"
      default_prefix="*186" usage_samples="" description="Activate privacy mode for current call."/>
    <feature_code ss_name="Privacy" feature_key="deactivate" feature="use" prefix="#186"
      default_prefix="#186" usage_samples="" description="Deactivate privacy mode for current call."/>
  </result>
</out>

```

[Hc_ss_incall_feature_code_set](#)

Команда используется для изменения списка feature кодов для работы с incall услуг в рамках определенного домена.

Аналог команды в CoCon:

```
/domain/<DOMAIN>/ss/feature-codes/incall/set
```

Метод http запроса:

POST

Шаблон http запроса:

`http://host:port/commands/ss_incall_feature_code_set`

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set supplementary services' incall feature codes for specific domain.
      Usage: http://server:9999/commands/ss_feature_code_set
      Copyright (c) 2016, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <!-- non-empty token -->
  <xs:simpleType name="prefixType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
  <!-- featureCodeType -->
  <xs:complexType name="featureCodeType">
    <xs:attribute name="ss_name" type="xs:string" use="required"/>
    <xs:attribute name="feature_key" type="xs:string" use="required"/>
    <xs:attribute name="feature" type="xs:string" use="optional"/>
    <xs:attribute name="prefix" type="prefixType" use="required"/>
    <xs:attribute name="default_prefix" type="prefixType" use="optional"/>
    <xs:attribute name="usage_samples" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- okType -->
  <xs:complexType name="okType"/>
  <!-- duplicatesType -->
  <xs:complexType name="duplicatesType">
    <xs:sequence>
      <xs:element name="feature_code" type="featureCodeType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="feature_code" type="featureCodeType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="result">
        <xs:complexType>
          <xs:choice>
            <xs:element name="ok" type="okType"/>
            <xs:element name="duplicates" type="duplicatesType"/>
          </xs:choice>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1

Запрос: http://192.168.1.21:9999/commands/ss_incall_feature_code_set

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local">
    <feature_code ss_name="CallRecord" feature_key="use" prefix="*111"/>
    <feature_code ss_name="Flash" feature_key="use" prefix="#222"/>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_incall_feature_code_set.xsd">
  <result>
    <ok/>
  </result>
</out>

```

Пример 2

Запрос:

http://192.168.1.21:9999/commands/ss_incall_feature_code_set

```
<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local">
    <feature_code ss_name="CallRecord" feature_key="use" prefix="*111"/>
    <feature_code ss_name="Flash" feature_key="use" prefix="*111"/>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_incall_feature_code_set.xsd">
  <result>
    <duplicates>
      <feature_code ss_name="CallRecord" feature_key="use" feature="use" prefix="*111"
        default_prefix="*111" usage_samples="" description="Start call record functionality."/>
      <feature_code ss_name="Flash" feature_key="use" feature="use" prefix="*111"
        default_prefix="*111" usage_samples="" description="Hold/Hole functionality."/>
    </duplicates>
  </result>
</out>
```

Hc_ss_license_packages - Команды управления пакетами лицензий

- [Hc_ss_licence_package_allocate](#)
- [Hc_ss_licence_package_free](#)
- [Hc_ss_licence_package_subscribers_list](#)

[Hc_ss_licence_package_allocate](#)

Команда аллокации лицензии ДВО на абоненте

Аналог команды в CoSop:

domain/<DOMAIN>/ss/licence/allocate

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_licence_package_allocate

Код HTTP-ответа:

- 200 – в случае успеха;

- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="package" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- warningType -->
  <xs:complexType name="warningType">
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="package" type="xs:string" use="required"/>
    <xs:attribute name="addresses" type="xs:string" use="required"/>
    <xs:attribute name="priority" type="xs:boolean"/>
  </xs:complexType>

  <!-- in -->

  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType" minOccurs="1"
maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->

  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok" type="okType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```

                <xs:element name="warning" type="warningType"
minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="error" type="errorType" minOccurs="1"
maxOccurs="1" />
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/ss_licence_package_allocate

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_ss_licence_package_allocate.xsd">
    <request domain="biysk.local" addresses="{240101, 240003, 240462}"
package="ECSS-ADV"/>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_ss_licence_package_allocate.xsd">
    <result>
        <ok address="240462" package="ECSS-ADV"/>
        <ok address="240003" package="ECSS-ADV"/>
        <ok address="240101" package="ECSS-ADV"/>
        <ok address="240003" package="ECSS-ADV"/>
        <ok address="240101" package="ECSS-ADV"/>
    </result>
</out>

```

hc_ss_licence_package_free

Команда освобождения лицензии ДВО на абоненте

Аналог команды в CoCon:

domain/<DOMAIN>/ss/licence/free

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_licence_package_free

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="package" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- warningType -->
  <xs:complexType name="warningType">
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="package" type="xs:string" use="required"/>
    <xs:attribute name="addresses" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->

  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType" minOccurs="1"
maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->

  <xs:element name="out">
    <xs:complexType>
```

```

        <xs:choice>
            <xs:element name="result" minOccurs="1" maxOccurs="1">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="ok" type="okType" minOccurs="0"
maxOccurs="unbounded"/>
                        <xs:element name="warning" type="warningType"
minOccurs="0" maxOccurs="unbounded"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <xs:element name="error" type="errorType" minOccurs="1"
maxOccurs="1" />
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/ss_licence_package_free

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_ss_licence_package_free.xsd">
    <request domain="biysk.local" addresses="{240003, 240462, 240500,
240505}" package="ECSS-ADV"/>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_ss_licence_package_free.xsd">
    <result>
        <ok address="240500" package="ECSS-ADV"/>
        <ok address="240462" package="ECSS-ADV"/>
        <ok address="240003" package="ECSS-ADV"/>
        <warning address="240505" reason="package_not_allocated"/>
    </result>
</out>

```


Нс_ss_licence_package_subscribers_list

Команда просмотра текущего распределения лицензий ДВО по доменам

Аналог команды в CoCon:

```
cluster/storage/<CLUSTER>/ss/limit
domain/<DOMAIN>/ss/limi
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_licence_package_subscribers_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="nameType">
    <xs:attribute name="name" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="packageIdType">
    <xs:attribute name="domain" type="xs:string"/>
    <xs:attribute name="name" type="xs:string"/>
    <xs:attribute name="address" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:element name="package_id" type="packageIdType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="cursor" type="xs:string"/>
  </xs:complexType>

  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:choice>
      <xs:element name="domain" type="nameType"/>
      <xs:element name="cluster" type="nameType"/>
    </xs:choice>
  </xs:complexType>

```

```

        </xs:choice>
        <xs:attribute name="package" type="xs:string"/>
        <xs:attribute name="limit" type="xs:string"/>
        <xs:attribute name="cursor" type="xs:string"/>
    </xs:complexType>

    <!-- in -->

    <xs:element name="in">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="request" type="requestType" minOccurs="1"
maxOccurs="1"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>

    <!-- out -->

    <xs:element name="out">
        <xs:complexType>
            <xs:choice>
                <xs:element name="result" type="resultType" minOccurs="0"
maxOccurs="unbounded"/>
                <xs:element name="error" type="errorType"/>
            </xs:choice>
        </xs:complexType>
    </xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/ss_licence_package_subscribers_list

```

<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_ss_licence_package_subscribers_list.xsd">
    <request package="ECSS-ADV" limit="5">
        <cluster name="ds1"/>
    </request>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_ss_licence_package_subscribers_list.xsd">
  <result
    cursor="g2gKZAANbw51c21hX3N1bGVjdGQAKnJ0b3AtY29uZm1ndXJhdGlvbi1scG0tc3MtcHJvZ
    mlsZXMtYWxpYXNlc2gCZAALYXN5bmNfZGlydHlnZAAJZHMXQGVjc3MxAAAAPAAAAAEBZAAJZHMXQG
    Vjc3MxZAALZGlzY19jb3BpZXNoBnIAA2QACWRzMUB1Y3NzMQEAA4FfTW4AA1Y256NheGEFcGADZAA
    JZHMXQGVjc3MxAQADCS1NfgACVjbn02phAGpkAA11bmRlZm1uZWZRkAA11bmRlZm1uZWRSAAAAAWgD
    aAhkAB1ydG9wLWxwbS1zcy1wcm9maWxlLWFsaWZlXJvd2QAAV9kAAFFZAABX2QACEVDU1MtQURWZ
    AAEdHJ1ZWQAAV9kAAFFamwAAAABZAACJF9qag==">
    <package_id domain="biysk.local" address="240503" name="ECSS-ADV"/>
    <package_id domain="biysk.local" address="240465" name="ECSS-ADV"/>
    <package_id domain="biysk.local" address="240501" name="ECSS-ADV"/>
    <package_id domain="biysk.local" address="240101" name="ECSS-ADV"/>
    <package_id domain="biysk.local" address="240502" name="ECSS-ADV"/>
  </result>
</out>
```

Нс_ss_package_limit - Команда для просмотра использования лицензий ДВО

Нс_ss_package_limit

Команда для просмотра текущего использования лицензий ДВО в рамках лицензионных профилей ДВО.

Аналог команды в CoCon:

```
domain/<DOMAIN>/ss/limit
cluster/storage/<DS_CLUSTER>/ss/limit
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_package_limit

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки (с указанием ошибки в теле ответа).

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="packageType">
    <xs:attribute name="name" type="xs:string"/>
    <xs:attribute name="domain_allocated" type="xs:string"/>
    <xs:attribute name="used" type="xs:integer"/>
    <xs:attribute name="available" type="xs:string"/>
    <xs:attribute name="licence_limit" type="xs:string"/>
    <xs:attribute name="description" type="xs:string"/>
    <xs:attribute name="ss_list" type="xs:string"/>
    <xs:attribute name="disabled_alias_list" type="xs:string"/>
    <xs:attribute name="priority_alias_list" type="xs:string"/>
    <xs:attribute name="alias_list" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="resultType">
    <xs:sequence maxOccurs="unbounded" minOccurs="0">
      <xs:element name="package" type="packageType" maxOccurs="unbounded" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="nameType">
    <xs:attribute name="name" type="xs:string"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:choice>
              <xs:element name="domain" type="nameType"/>
              <xs:element name="cluster" type="nameType"/>
            </xs:choice>
            <xs:attribute name="show_ss" type="xs:boolean"/>
            <xs:attribute name="show_aliases" type="xs:boolean"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

        <xs:element name="error" type="errorType"/>
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Примеры

Запрос:

http://192.168.1.21:9999/commands/ss_package_limit

```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_package_limit.xsd">
  <request>
    <domain name="biysk.local"/>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_package_limit.xsd">
  <result>
    <package name="ECSS-GEN" used="0" available="infinity" domain_allocated="infinity"
description="Стандартные услуги (РД 45)"/>
    <package name="ECSS-BAS" used="0" available="infinity" domain_allocated="infinity"
description="Базовые услуги (РД 45)"/>
    <package name="ECSS-BAS+" used="1" available="infinity" domain_allocated="infinity"
description="Расширение базовых услуг"/>
    <package name="ECSS-ADV" used="2" available="infinity" domain_allocated="infinity"
description="Дополнительные услуги"/>
  </result>
</out>

```

Если лицензия не включает пакеты услуг:

```

<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_package_limit.xsd">
  <error cmd="hc_ss_package_limit" reason="licence_not_used_packages"/>
</out>

```

`Hc_ss_preset_packages_info` - Команда, возвращающая список ДВО с указанием их уровня

- [Hc_ss_preset_packages_info](#)

`Hc_ss_preset_packages_info`

Команда позволяет вывести список ДВО с указанием их уровня.

Аналог команды в CoCon:

Нет

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_preset_packages_info

Код HTTP-ответа:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
  <xs:complexType name="presetPackageType">
    <xs:attribute name="level" type="xs:integer"/>
    <xs:attribute name="ss_list" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:element name="preset_package" type="presetPackageType"
minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="result" type="resultType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/ss_preset_packages_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss_preset_packages_info.xsd">
  <request />
</in>

```

Ответ: 200


```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_preset_packages_info.xsd">
  <result>
    <preset_package level="1"
  ss_list="cfb,cfnr,cfos,cfu,clip,cnip,choold,ctr,cw,direct_call,mcid,cc_agent,t
  eleconference,teleconference_manager,tsmn_request,qsig_cfb,qsig_cfnr,qsig_cfu
  "/>
    <preset_package level="2"
  ss_list="3way,clir,cliro,dnd,pickup,redial,cfnr_type2,cfu_type2,m1"/>
    <preset_package level="3"
  ss_list="acb,alarm,auto_redial,auto_redial_with_callback,callback,cf_sip,conf
  erence,fbc,hide_cf_name,meet_me_conference,rbp,rfc,ring_back_tone,sca,sco_bla
  ck,sco_white,scr,speed_dial,voicemail,vip_call,cf_aon"/>
    <preset_package level="4"
  ss_list="call_recording,blf,cft,cgg,chunt,click_to_call,fax_receiver,fax_to_e
  mail,find_me,find_me_no_response,follow_me,follow_me_no_response,intercom,par
  k,presence,second_handset,sip_message"/>
    <preset_package level="5"
  ss_list="avatar,assistant,boss_group,cpip,distinctive_picture,distinctive_rin
  g,flexicall,fxo_trunk,intervention,introduce,message,multicast_listen,multica
  st_page,personal_ivr,privacy,remote_phone,remote_ring,smart_cancel,voice_page
  ,voice_page_control,walkie_talkie,zone_page"/>
  </result>
</out>

```

HC_ss_system_profile - Команды управления системным профилем

- [HC_ss_system_profile_copy](#)
- [HC_ss_system_profile_copy_to_domain](#)
- [HC_ss_system_profile_info](#)
- [HC_ss_system_profile_list](#)
- [HC_ss_system_profile_remove](#)
- [HC_ss_system_profile_set](#)
- [HC_ss_system_profile_add_ss](#)

HC_ss_system_profile_copy

Команда позволяет сделать копию системного профиля с новым именем.

Аналог команды в CoCon:

cluster/storage/<STORAGE>/ss/profile/copy

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_system_profile_copy

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки (с указанием ошибки в теле ответа).

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="storage" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Data Storage cluster's name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="src_name" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Source profile name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="dst_name" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Destination profile name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1

Запрос: http://192.168.1.21:9999/commands/ss_system_profile_copy

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_system_profile_copy.xsd">
  <request storage="ds1" src_name="ss1" dst_name="ss_test"/>
</in>
```

Ответ: 201

Пример 2

В случае, если профиль с именем profile3 уже существует

Запрос: http://192.168.23.34:9999/commands/ss_system_profile_copy

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_system_profile_copy.xsd">
  <request storage="ds1" src_name="ss1" dst_name="ss_test"/>
</in>
```

Ответ: 200

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_system_profile_copy.xsd">
  <error cmd="hc_ss_system_profile_copy" reason="already_exists" body="{error,already_exists}"/>
  >
</out>
```

Пример 3

В случае, если исходный профиль не существует

Запрос: http://192.168.23.34:9999/commands/ss_system_profile_copy

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_system_profile_copy.xsd">
  <request storage="ds1" src_name="unknown_profile" dst_name="to"/>
</in>
```

Ответ: 200

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_system_profile_copy.xsd">
  <error cmd="hc_ss_system_profile_copy" reason="not_exists" body="{error,not_exists}"/>
</out>
```

hc_ss_system_profile_copy_to_domain

Команда позволяет скопировать системный профиль услуги в домен с возможностью изменения имени профиля.

Аналог команды в CoCon:

`c\cluster/storage/<STORAGE>/ss/profile/copy-to-domain`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/ss_system_profile_copy_to_domain`

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки (с указанием ошибки в теле ответа).

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- copyType -->
  <xs:complexType name="copyType">
    <xs:attribute name="system_profile" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          System profile name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="domain" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Destination domain name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="domain_profile" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Destination domain profile name. In case of domain_profile doesn't set,
          domain_profile = system_profile.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="copy" type="copyType" minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="storage" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Data Storage cluster's name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1

Запрос: http://192.168.1.21:9999/commands/ss_system_profile_copy_to_domain

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_system_profile_copy_to_domain.xsd">
  <request storage="ds1">
    <copy system_profile="ss_test" domain="biysk.local"/>
  </request>
</in>

```

Ответ:

201

Пример 2

В случае, если системного профиля sprofile2.2 не существует, а так же домена domain2 не существует.

Запрос: http://192.168.23.34:9999/commands/ss_system_profile_copy_to_domain

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_system_profile_copy_to_domain.xsd">
  <request storage="ds1">
    <copy system_profile="sprofile2" domain="domain2" domain_profile="dprofile2.2"/>
  </request>
</in>

```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_system_profile_copy_to_domain.xsd">
  <error cmd="hc_ss_system_profile_copy_to_domain" reason="domain_not_exists"
  body="domain_not_exists" entity="domain2"/>
</out>
```

При этом профиль profile3 был успешно скопирован в домен biysk.local.

HC_ss_system_profile_info

Команда позволяет получить настройки системного профиля услуги по имени.

Аналог команды в CoCon:

ccluster/storage/<STORAGE>/ss/profile/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_system_profile_info

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки (с указанием ошибки в теле ответа).

XSD-схема XML-файла запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="ss.xsd"/>
  <xs:include schemaLocation="error.xsd"/>
  <!-- ssInfoType -->
  <xs:complexType name="ssInfoType">
    <xs:complexContent>
      <xs:extension base="ssType">
        <xs:attribute name="available" type="xs:boolean" use="required">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              Include or not current supplementary service in the profile.
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- profileType -->
  <xs:complexType name="profileType">
    <xs:sequence>
      <xs:element name="ss" type="ssInfoType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          System's profile name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="description" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Human readable description of the current profile.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="storage" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Data Storage cluster's name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="profile" type="xs:string" use="required">

```

```

    <xs:annotation>
      <xs:documentation xml:lang="en">
        Profile name to show information only about it.
      </xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request" type="requestType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="profile" type="profileType"/>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/ss_system_profile_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ss_system_profile_info.xsd">
  <request storage="ds1" profile="profile1"/>
</in>

```

Ответ:

200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_system_profile_info.xsd">
  <profile name="ss_biysk" description="ss_tstpfofile">
    <ss
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="multicast_listen" version="1"
      description="Multicast Listen Service" active="false" enabled="false" available="false">
      <msd>
        <property name="enumCodec" type="string" description="Internal field to codec
property" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|G726-40"/>
        <property name="integer_rangePort" type="string" description="Internal field
to udp port property" value="[10000,65535]"/>
      </msd>
      <csd>
        <property name="ip" type="ip" description="Ip address for listen multicast."/>
        <property name="port" type="integer_range" description="Port for listen
multicast (10000-65535)."/>
        <property name="codec" type="enum" description="Audio codec for multicast
traffic." value="G722"/>
      </csd>
    </ss>
    <ss
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="vip_call" version="2" description="Vip call"
      active="false" enabled="false" available="false">
      <msd>
        <property name="enumPriority" type="string" description="Call's priority"
value="0_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
      </msd>
      <csd>
        <property name="priority" type="enum" description="(0_flash_override | 1_flash
| 2_immediate | 3_priority | 4_routine)" value="4_routine"/>
      </csd>
    </ss>
    <ss
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="voice_page_control" version="2"
      description="'Voice Page Control' service" active="false" enabled="false" available="false">
      <csd>
        <property name="accept_incoming_vp" type="boolean" description="Accept or
Reject voice page calls" value="true"/>
        <property name="block_incoming_vp_by_dnd" type="boolean" description="Block
voice page calls by DND service" value="true"/>
        <property name="mute_incoming_vp" type="boolean" description="Accept incoming
voice page call in silent mode" value="true"/>
      </csd>
    </ss>
  </profile>
</out>

```

```

<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="redial" version="2" description="Last number
redial" active="false" enabled="false" available="false"/>
  <ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="ml" version="4" description="Multiline"
    active="false" enabled="false" available="false">
      <msd>
        <property name="integer_rangeLine_count" type="string"
description="Internal field" value="[0,16]"/>
        <property name="integer_rangeLine_count_out" type="string"
description="Internal field" value="[-1,16]"/>
      </msd>
      <csd>
        <property name="line_count" type="integer_range" description="Line count"
value="3"/>
        <property name="line_count_out" type="integer_range" description="Outgoing
lines count" value="-1"/>
      </csd>
    </ss>
  <ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="fxo_trunk" version="1" description="FX0 trunk
service" active="false" enabled="false" available="false">
      <msd>
        <property name="owner" type="atom" description="SS type. Available values:
user | trunk | any" value="trunk"/>
      </msd>
    </ss>
  <ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="sco_black" version="2" description="Selective
Call Originating, Black list" active="false" enabled="false" available="false">
      <csd>
        <property_list name="blacklist" type="index_address" maxlen="50"
description="Black list members. Format: <address>[/<index>]"/>
        <undefined/>
      </property_list>
    </csd>
  </ss>
  <ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="zone_page" version="9" description="Zone Page
call" active="false" enabled="false" available="false">
      <msd>
        <property name="enumPriority" type="string" description="Call's priority"
value="0_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
        <property name="enumInitiation_policy" type="string"
description="Initiation call's policy" value="managers|members|any"/>
      </msd>
    </ss>
  </ss>

```

```

        <property name="integer_rangeMulticast_port" type="string"
description="Internal field" value="[1,65535]"/>
        <property name="integer_rangeMulticast_ttl" type="string"
description="Internal field" value="[1,255]"/>
        <property name="enumMulticast_codec" type="string" description="Internal
field" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|G726-40"/>
        </msd>
        <ssd>
            <property name="dtmf_detector" type="boolean" description="" value="true"/>
        >
        </ssd>
        <csd>
            <property name="priority" type="enum" description="(0_flash_override |
1_flash | 2_immediate | 3_priority | 4_routine)" value="3_priority"/>
            <property name="intro_pause" type="integer" description="Pause before play
intro files" value="3"/>
            <property name="announcement_pause" type="integer" description="Pause
before play announcement files" value="0"/>
            <property name="announcement_playback_cnt" type="integer"
description="Announcement files playback count" value="1"/>
            <property name="announcement_repeat_time" type="integer"
description="Announcement files repeat timeout" value="10"/>
            <property name="terminate_when_initiator_disconnects" type="boolean"
description="Terminate call after initiator disconnected"/>
            <property name="terminate_when_playback_ends" type="boolean"
description="Terminate call after playback ends"/>
            <property name="terminate_when_last_listener_leave" type="boolean"
description="Terminate call after last listener disconnects"/>
            <property name="initiation_policy" type="enum" description="Initiation
call's policy" value="any"/>
            <property name="display_name" type="string" description="ZonePage call
DisplayName" value="ZonePage"/>
            <property name="play_release_tone" type="boolean" description="ZonePage
release tone" value="false"/>
            <property name="announcement_repeat_digit" type="address"
description="Digit to repeat announcement files" value="#" />
            <property name="multicast_ip" type="ip" description="Optional multicast
address to use for announcement" value="0.0.0.0"/>
            <property name="multicast_port" type="integer_range" description="Port of
multicast address" value="0"/>
            <property name="multicast_codec" type="enum" description="Codec to be used
for multicasting" value="PCMA"/>
            <property name="multicast_ttl" type="integer_range" description="TTL for
multicast" value="1"/>
            <property_list name="members" type="address" maxlen="256"
description="Zone page member list">
                <undefined/>
            </property_list>
            <property_list name="managers" type="address" maxlen="256"
description="Zone page managers list"/>

```

```

        <property_list name="intro_files" type="media_resource" maxlen="10"
description="List of Intro media_files"/>
        <property_list name="announcement_files" type="media_resource" maxlen="10"
description="List of Announcement media_files"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="meet_me_conference" version="2" description="Meet
Me Conference" active="false" enabled="false" available="false">
    <msd>
        <property name="regimeMode" type="string" description="Internal field to
validate mode property" value="master:destroy_mode:room_number|member:room_number"/>
        <property name="enumDestroy_mode" type="string" description="Internal
field to validate mode property" value="by_no_master|by_no_more_calls"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
        </property_list>
    </msd>
    <csd>
        <property name="mode" type="regime" description="The mode of meet me
conference (master|member)"/>
        <property name="destroy_mode" type="enum" description="The mode of
destroying conference (by_no_master|by_no_more_calls)" value="by_no_more_calls"/>
        <property name="room_number" type="address" description="Room number"
value=""/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="teleconference" version="4"
description="Teleconference" active="false" enabled="false" available="false">
    <msd>
        <property name="enumType" type="string" description="Teleconference type:
meeting or room" value="meeting|room"/>
    </msd>
    <csd>
        <property name="destroy_by_no_initiator" type="boolean"
description="Destroy the teleconference by release from the initiator?" value="true"/>
        <property name="template_from_room" type="teleconference_room"
description="Get template from the teleconference room" value=""/>
        <property name="type" type="enum" description="Teleconference type:
meeting or room" value="meeting"/>
    </csd>
</ss>
<ss

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="mcid" version="2" description="Malicious Call
Identification" active="false" enabled="false" available="false"/>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="assistant" version="2" description="Chief-
assistant group" active="false" enabled="false" available="false">
        <msd>
            <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="user"/>
        </msd>
        <ssd>
            <property name="forwarding_ss" type="boolean" description=""
value="true"/>
        </ssd>
        <csd>
            <property_list name="assistants" type="index_address" maxlen="4"
description="Assistants numbers. Format: <address>[/<index>]">
                <undefined/>
            </property_list>
            <property_list name="whitelist" type="index_address" maxlen="128"
description="White list members. Format: <address>[/<index>]">
                <undefined/>
            </property_list>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="ctr" version="5" description="Call Transfer"
active="false" enabled="false" available="false">
        <msd>
            <property_list name="dependency_list" type="atom" maxlen="0"
description="List of SS that is depended from SS">
                <item value="chold"/>
            </property_list>
        </msd>
        <ssd>
            <property name="dtmf_detector" type="boolean" description=""
value="true"/>
        </ssd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="personal_ivr" version="2" description="Personal
IVR script" active="false" enabled="false" available="false">
        <msd>
            <property name="handle_module" type="atom" description=""
value="ss_personal_ivr_handler"/>
            <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="user"/>

```

```

        </msd>
        <csd>
            <property name="script" type="ivr_script_id" description="Script ID"
value=""/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="voicemail" version="12" description="Voice mail
service" active="false" enabled="false" available="false">
        <msd>
            <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
                <item value="assistant"/>
                <item value="follow_me"/>
                <item value="cfu_type2"/>
                <item value="meet_me_conference"/>
                <item value="cfu"/>
                <item value="find_me"/>
                <item value="cf_aon"/>
                <item value="cft"/>
            </property_list>
        </msd>
        <ssd>
            <property name="dtmf_detector" type="boolean" description=""
value="true"/>
        </ssd>
        <csd>
            <property name="no_reply" type="boolean" description="Activate service
when subscriber no reply" value="true"/>
            <property name="busy" type="boolean" description="Activate service
when subscriber is busy" value="false"/>
            <property name="out_of_service" type="boolean" description="Activate
service when subscriber is out of service" value="true"/>
            <property name="unconditional" type="boolean" description="Activate
service any time" value="false"/>
            <property name="timeout" type="positive_integer" description="Timeout
before call forward (sec)" value="30"/>
            <property name="send_by_email" type="boolean" description="Send
recorded message by email" value="false"/>
            <property name="min_message_duration" type="positive_integer"
description="The minimum voice mail duration (sec). In case of voice mail duration will be
less that min_message_duration, this message will be dropped" value="0"/>
            <property name="max_message_duration" type="positive_integer"
description="The maximum voice mail duration (sec). 0 - in case of not limited duration"
value="300"/>
            <property name="mailbox_volume" type="positive_integer"
description="The mailbox volume (sec)" value="1800"/>
            <property name="override_messages" type="boolean"
description="Override messages in case of mailbox is full" value="true"/>

```



```

        <property name="email_from" type="string" description="Email from
template. Macro variables: %DATE%, %TIME%, %CALLING%, %DISPLAYNAME%" value="%DISPLAYNAME%"/>
        <property name="email_subject" type="string" description="Email
subject template. Macro variables: %DATE%, %TIME%, %CALLING%, %DISPLAYNAME%" value="Voicemail
from %CALLING% at %DATE% %TIME%"/>
        <property name="email_body" type="string" description="Email body
template. Macro variables: %DATE%, %TIME%, %CALLING%, %DISPLAYNAME%" value=""/>
        <property name="play_message_details" type="boolean" description="Play
message's details (number, time) before message" value="false"/>
        <property name="password" type="address" description="There is
password to have access to the voicemail box" value=""/>
        <property name="max_silence" type="positive_integer"
description="There is time(sec) while service wait for a contiguous period of silence before
terminating an incoming call to voice mail" value="0"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="auto_redial" version="2" description="Auto
redial" active="false" enabled="false" available="false">
    <csd>
        <property name="recall_timeout" type="positive_integer"
description="Timeout between recalls (sec)" value="15"/>
        <property name="recall_count" type="positive_integer"
description="Recalls' count" value="6"/>
        <property name="no_answer_timeout" type="positive_integer"
description="Awaiting answer timeout (sec)" value="30"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="click_to_call" version="5" description="Click to
call" active="false" enabled="false" available="false">
    <csd>
        <property name="number" type="address" description="User's number"/>
        <property name="auto_answer" type="boolean" description="Auto answer"
value="true"/>
        <property_list name="hashes" type="string" maxlen="10"
description="User's hashes">
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="distinctive_ring" version="1"
description="Destination ringtone change" active="false" enabled="false" available="false">
    <csd>
        <property_list name="filters" type="distinctive_ring" maxlen="16"
description="Distinctive ring for specific numbers">

```

```

        <undefined/>
    </property_list>
</csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="cw" version="5" description="Call Waiting"
  active="false" enabled="false" available="false">
  <msd>
    <property_list name="dependency_list" type="atom" maxlen="0"
description="List of SS that is depended from SS">
      <item value="chold"/>
    </property_list>
  </msd>
  <ssd>
    <property name="dtmf_detector" type="boolean" description=""
value="true"/>
  </ssd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="call_recording" version="10" description="Call
recording" active="false" enabled="false" available="false">
  <msd>
    <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="any"/>
    <property name="enumMode" type="string" description="Internal field to
validate mode property" value="always_on|on_demand|after_answer"/>
  </msd>
  <ssd>
    <property name="locked" type="boolean" description="Can subscriber
manage current SS, or not" value="true"/>
    <property name="dtmf_detector" type="boolean" description=""
value="true"/>
  </ssd>
</csd>
  <property name="mode" type="enum" description="The mode of the
recording call (always_on|on_demand)"/>
</csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="remote_ring" version="1" description="Remote ring
to callee" active="false" enabled="false" available="false">
  <msd>
    <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
      <item value="distinctive_ring"/>
    </property_list>
  </msd>

```

```

        <csd>
            <property_list name="announcement_files" type="media_resource"
maxlen="10" description="List of Announcement media_files"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="blf" version="3" description="Busy Lamp Field"
active="false" enabled="false" available="false">
        <csd>
            <property name="max_subscribe" type="positive_integer"
description="Maximum number of subscriptions" value="16"/>
            <property name="ignore_page_calls" type="boolean" description="Doesn't
send BLF notification from Zone Page calls" value="true"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="remote_phone" version="1" description="Remote
phone" active="false" enabled="false" available="false">
        <msd>
            <property name="handle_module" type="atom" description=""
value="ss_remote_phone_handler"/>
        </msd>
        <csd>
            <property name="remote_endpoint" type="interface" description="Remote
interface" value=""/>
            <property name="pin" type="address" description="PIN-code"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="presence" version="2" description="SIP Presence"
active="false" enabled="false" available="false">
        <csd>
            <property name="max_subscribe" type="positive_integer"
description="Maximum number of subscriptions" value="16"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="message" version="2" description="Message
service" active="false" enabled="false" available="false">
        <msd>
            <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="any"/>
        </msd>
        <ssd>
            <property name="message_box_size" type="integer" description="Message
box size" value="20"/>

```

```

        <property name="remove_only_after_answer" type="boolean"
description="Remove message only after abonent answer" value="false"/>
    </ssd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="ring_back_tone" version="2" description="Ring
back tone" active="false" enabled="false" available="false">
    <csd>
        <property name="file" type="media_resource" description="File to play
on ring"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="voice_page" version="9" description="Voice Page
call" active="false" enabled="false" available="false">
    <msd>
        <property name="enumPriority" type="string" description="Call's
priority" value="0_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
    </msd>
    <ssd>
        <property name="dtmf_detector" type="boolean" description=""
value="true"/>
    </ssd>
    <csd>
        <property name="skip_cf" type="boolean" description="Skipping call
forwarding" value="true"/>
        <property name="priority" type="enum" description="(0_flash_override |
1_flash | 2_immediate | 3_priority | 4_routine)" value="3_priority"/>
        <property name="announcement_repeat_time" type="integer"
description="Announcement files repeat timeout" value="10"/>
        <property name="announcement_playback_cnt" type="integer"
description="Announcement files playback count [0..5]" value="0"/>
        <property name="swap_digit" type="address" description="Digit to swap
reception / transmission mode" value="*/>
        <property name="announcement_repeat_digit" type="address"
description="Digit to repeat announcement files" value="#"/>
        <property_list name="intro_files" type="media_resource" maxlen="10"
description="List of Intro media_files"/>
        <property_list name="announcement_files" type="media_resource"
maxlen="10" description="List of Announcement media_files"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="chold" version="8" description="Call hold"
active="false" enabled="false" available="false">
    <ssd>

```

```

        <property name="dtmf_detector" type="boolean" description=""
value="true"/>
    </ssd>
    <csd>
        <property name="dtmf_sequence_as_flash" type="boolean"
description="Try interpret the DTMF sequence as a flash (look at incall feature codes)"
value="false"/>
        <property name="disable_moh" type="boolean" description="Disable music
on hold" value="false"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="cfos" version="12" description="Call Forwarding
Out of Service" active="false" enabled="false" available="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="any"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="tsmn_request"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean" description=""
value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address" description="Forward number"/>
        <property name="external_number" type="address" description="External
forward number" value=""/>
        <property name="use_external_number" type="boolean" description="Use
external forward number" value="false"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="scr" version="2" description="Selective Call
Rejection" active="false" enabled="false" available="false">
    <csd>
        <property_list name="blacklist" type="index_address" maxlen="50"
description="Black list members. Format: <address>[/<index>]">
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="boss_group" version="1" description="Boss-group"
active="false" enabled="false" available="false">
    <msd>
        <property name="integer_rangeDelay_ring_timeout" type="string"
description="Internal field" value="[0,60]"/>
        <property name="integer_rangeLines_count" type="string"
description="Internal field" value="[0,16]"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="cw"/>
        </property_list>
        <property_list name="dependency_list" type="atom" maxlen="0"
description="List of SS that is depended from SS">
            <item value="auto_redial"/>
        </property_list>
    </msd>
    <ssd>
        <property name="dtmf_detector" type="boolean" description=""
value="true"/>
    </ssd>
    <csd>
        <property name="display_name" type="string" description="Display name
for boss group" value=""/>
        <property name="delay_ring_timeout" type="integer_range"
description="Timeout before call on delayed members" value="0"/>
        <property name="lines_count" type="integer_range"
description="Incoming boss-group lines count" value="2"/>
        <property name="members_fwd_busy_override" type="boolean"
description="Ignore members CFB for group calls" value="true"/>
        <property name="call_waiting_tone" type="boolean" description="Play
call waiting tone for second(s) calls on group" value="false"/>
        <property_list name="members" type="address" maxlen="8"
description="Members for immediate call">
            <undefined/>
        </property_list>
        <property_list name="delay_members" type="address" maxlen="8"
description="Members for delay call">
            <undefined/>
        </property_list>
        <property_list name="silent_members" type="address" maxlen="8"
description="Members for BLF and pickup only group calls">
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cf_aon" version="8" description="Call Forwarding
AON" active="false" enabled="false" available="false">

```

```

        <msd>
            <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="any"/>
            <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
                <item value="assistant"/>
                <item value="follow_me"/>
                <item value="cfu_type2"/>
                <item value="meet_me_conference"/>
                <item value="cfu"/>
                <item value="find_me"/>
                <item value="cf_aon"/>
            </property_list>
        </msd>
        <ssd>
            <property name="forwarding_ss" type="boolean" description=""
value="true"/>
        </ssd>
        <csd>
            <property name="no_reply" type="boolean" description="Activate service
when subscriber no reply" value="false"/>
            <property name="timeout" type="positive_integer" description="Timeout
before call forward (sec)" value="15"/>
            <property name="number" type="address" description="Forward number"/>
            <property_list name="prefix_number" type="address" maxlen="10"
description="Prefix number A">
                <undefined/>
            </property_list>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="intercom" version="1" description="Intercom"
active="false" enabled="false" available="false"/>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="direct_call" version="2" description="Direct
call" active="false" enabled="false" available="false">
                <csd>
                    <property name="number" type="address" description="Call number to
direct call"/>
                    <property name="timeout" type="positive_integer"
description="Timeout before direct call (sec)" value="5"/>
                </csd>
            </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="privacy" version="2" description="Privacy"
active="false" enabled="false" available="false">
        </msd>

```

```

        <property name="enumMode" type="string" description="Mode"
value="on_demand|always_on"/>
    </msd>
    <ssd>
        <property name="dtmf_detector" type="boolean" description=""
value="true"/>
    </ssd>
    <csd>
        <property name="mode" type="enum" description="Privacy usage mode"
value="on_demand"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cgg" version="7" description="Group-calling"
active="false" enabled="false" available="false">
    <msd>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
            <item value="cfu"/>
            <item value="find_me"/>
            <item value="cf_aon"/>
            <item value="cft"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean" description=""
value="true"/>
    </ssd>
    <csd>
        <property name="send_call_diversion" type="boolean"
description="Send call diversion flag in call signalisation" value="true"/>
        <property_list name="numbers" type="address" maxlen="10"
description="Group's numbers">
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cfu" version="14" description="Call Forwarding
Unconditional" active="false" enabled="false" available="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="any"/>

```



```

        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean" description=""
value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address" description="Forward
number"/>
        <property name="external_number" type="address"
description="External forward number" value=""/>
        <property name="use_external_number" type="boolean"
description="Use external forward number" value="false"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="conference" version="9" description="Conference
Call, Add-on" active="false" enabled="false" available="false">
    <msd>
        <property name="enumDestroy_mode" type="string"
description="Internal field to validate mode property" value="by_no_master|by_no_more_calls"/>
        <property_list name="dependency_list" type="atom" maxlen="0"
description="List of SS that is depended from SS">
            <item value="chold"/>
            <item value="ctr"/>
        </property_list>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="3way"/>
        </property_list>
    </msd>
    <csd>
        <property name="destroy_mode" type="enum" description="The mode of
destroying conference (by_no_master|by_no_more_calls)" value="by_no_master"/>
        <property name="max_participants" type="positive_integer"
description="The maximum number of participants in the conference, limited to the number 64
at the top. By default 16" value="16"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="find_me" version="7" description="Find Me"
    active="false" enabled="false" available="false">

```

```

        <msd>
            <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
                <item value="assistant"/>
                <item value="follow_me"/>
                <item value="cfu_type2"/>
                <item value="meet_me_conference"/>
                <item value="cfu"/>
            </property_list>
        </msd>
    </ssd>
    <property name="forwarding_ss" type="boolean" description=""
value="true"/>
    </ssd>
    <csd>
        <property name="send_call_diversion" type="boolean"
description="Send call diversion flag in call signalisation" value="true"/>
        <property_list name="groups" type="find_me_group" maxlen="32"
description="Find me groups"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfu" version="3" description="QSIG: Call
Forwarding Unconditional" active="false" enabled="false" available="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="trunk"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="follow_me"/>
        </property_list>
    </msd>
    <csd>
        <property name="hide_forwarding" type="boolean" description="Hide
forwarding fields (rgn, ocdrn)" value="false"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="follow_me" version="11" description="Follow me"
    active="false" enabled="false" available="false">
    <msd>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="assistant"/>
        </property_list>
    </msd>
</ssd>

```

```

        <property name="forwarding_ss" type="boolean" description=""
value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address" description="Followed
number" value=""/>
        <property name="pin" type="address" description="PIN-code"
value=""/>
        <property name="external_number" type="address"
description="External forward number" value=""/>
        <property name="use_external_number" type="boolean"
description="Use external forward number" value="false"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="walkie_talkie" version="1" description="Walkie
Talkie" active="false" enabled="false" available="false">
    <ssd>
        <property name="dtmf_detector" type="boolean" description=""
value="true"/>
    </ssd>
    <csd>
        <property name="swap_digit" type="address" description="Digit to
swap reception / transmission mode" value="*/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="multicast_page" version="1"
description="Multicast IP Paging" active="false" enabled="false" available="false">
    <msd>
        <property name="enumCodec" type="string" description="Internal
field" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|G726-40"/>
        <property name="integer_rangePort" type="string"
description="Internal field" value="[10000,65535]"/>
        <property name="integer_rangeTtl" type="string"
description="Internal field" value="[1,255]"/>
    </msd>
    <csd>
        <property name="ip" type="ip" description="IP address to cast the
media to"/>
        <property name="port" type="integer_range" description="UDP port
to cast the media to (10000-65535)"/>
        <property name="codec" type="enum" description="Codec to be used
for multicasting" value="PCMA"/>
        <property name="ttl" type="integer_range" description="TTL for
multicast packets (1-255)" value="1"/>
    </csd>
</ss>

```

```

<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="cft" version="14" description="Call Forwarding
Unconditional Time Dependent" active="false" enabled="false" available="false">
  <msd>
    <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="any"/>
    <property name="enumTime_format" type="string"
description="Internal field to validate time_format property" value="local|utc"/>
    <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
      <item value="assistant"/>
      <item value="follow_me"/>
      <item value="cfu_type2"/>
      <item value="meet_me_conference"/>
      <item value="cfu"/>
      <item value="find_me"/>
      <item value="cf_aon"/>
    </property_list>
  </msd>
  <ssd>
    <property name="forwarding_ss" type="boolean" description=""
value="true"/>
  </ssd>
  <csd>
    <property name="day_1" type="term"
description="[[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber]"
value="off"/>
    <property name="day_2" type="term"
description="[[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber]"
value="off"/>
    <property name="day_3" type="term"
description="[[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber]"
value="off"/>
    <property name="day_4" type="term"
description="[[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber]"
value="off"/>
    <property name="day_5" type="term"
description="[[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber]"
value="off"/>
    <property name="day_6" type="term"
description="[[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber]"
value="off"/>
    <property name="day_7" type="term"
description="[[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber]"
value="off"/>
    <property name="time_format" type="enum" description="CFT for
Sunday (local | utc)" value="local"/>
  </csd>
</ss>

```

```

<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="chunt" version="7" description="Call hunt groups"
  active="false" enabled="false" available="false">
  <msd>
    <property name="regimeMode" type="string" description="Internal
field to validate search_strategy property" value="group:numbers:queue_strategy|
serial:numbers:cyclic:search_strategy>window_shift_timeout:max_shift_count:queue_strategy|
longest_idle:numbers:cyclic>window_shift_timeout:max_shift_count:queue_strategy|
delay_group:numbers>window_shift_timeout:queue_strategy|manual"/>
    <property name="enumSearch_strategy" type="string"
description="Internal field to validate search_strategy property" value="first|last|random|
longest_idle"/>
    <property name="enumQueue_strategy" type="string"
description="Internal field to validate queue_strategy property" value="wait|drop"/>
    <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
      <item value="assistant"/>
      <item value="follow_me"/>
      <item value="cfu_type2"/>
      <item value="meet_me_conference"/>
      <item value="cfu"/>
      <item value="find_me"/>
      <item value="cf_aon"/>
      <item value="cft"/>
      <item value="cgg"/>
    </property_list>
  </msd>
  <ssd>
    <property name="delay_retrieve_timeout_1" type="integer"
description="Delay retrieve timeout for release group call by group subscriber" value="1000"/>
  >
    <property name="delay_retrieve_timeout_2" type="integer"
description="Delay retrieve timeout for release group call by slave subscriber" value="5000"/>
  >
    <property name="delay_retrieve_timeout_3" type="integer"
description="Delay retrieve timeout for waiting free group subscriber" value="15000"/>
    <property name="forwarding_ss" type="boolean" description=""
value="true"/>
  </ssd>
  <csd>
    <property name="mode" type="regime" description="The type of the
chunt"/>
    <property name="cyclic" type="boolean" description="Flag that is
show is call hunt group cyclic or not" value="false"/>
    <property name="reset_window_start_pos_after_call" type="boolean"
description="Reset window start position after all incoming calls will be connected with
group calls" value="false"/>
    <property name="window_shift_timeout" type="positive_integer"
description="Interval after which window shift on the window_shift value" value="0"/>

```

```

        <property name="max_shift_count" type="integer" description="The
number of times that a window can be shift" value="-1"/>
        <property name="window_size" type="positive_integer"
description="The number of simultaneous calls to the call-hunt's subscribers" value="1"/>
        <property name="window_start_pos" type="integer" description="The
index (zero-based) of the first subscriber in the window" value="0"/>
        <property name="window_shift" type="positive_integer"
description="The value on which shift window_start_pos when window_shift_timeout fired"
value="1"/>
        <property name="queue_strategy" type="enum" description="(wait |
drop)" value="wait"/>
        <property name="search_strategy" type="enum" description="(first |
last | random | longest_idle)" value="first"/>
        <property name="queue_size" type="positive_integer"
description="Wait calls' queue size" value="0"/>
        <property_list name="numbers" type="address" maxlen="10"
description="Call-hunt member list">
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cfnr_type2" version="11" description="Call
Forwarding No Reply without notification" active="false" enabled="false" available="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="any"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="cfnr"/>
            <item value="follow_me"/>
            <item value="tsmn_request"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean" description=""
value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address" description="Forward
number"/>
        <property name="timeout" type="positive_integer"
description="Timeout before call forward (sec)" value="15"/>
        <property name="use_redirection_counter" type="boolean"
description="Use redirectionCounter field to count hops" value="false"/>
        <property name="external_number" type="address"
description="External forward number" value=""/>

```

```

        <property name="use_external_number" type="boolean"
description="Use external forward number" value="false"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="acb" version="2" description="Anonymous Call
Blocking" active="false" enabled="false" available="false"/>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="cfu_type2" version="14" description="Call
Forwarding Unconditional without notification" active="false" enabled="false"
available="false">
            <msd>
                <property name="owner" type="atom" description="SS type.
Available values: user | trunk | any" value="any"/>
                <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
                    <item value="assistant"/>
                    <item value="follow_me"/>
                </property_list>
            </msd>
            <ssd>
                <property name="forwarding_ss" type="boolean" description=""
value="true"/>
            </ssd>
            <csd>
                <property name="number" type="address" description="Forward
number"/>
                <property name="use_redirection_counter" type="boolean"
description="Use redirectionCounter field to count hops" value="false"/>
                <property name="external_number" type="address"
description="External forward number" value=""/>
                <property name="use_external_number" type="boolean"
description="Use external forward number" value="false"/>
            </csd>
        </ss>
    </ss>
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="cfnr" version="11" description="Call Forwarding
No Reply" active="false" enabled="false" available="false">
        <msd>
            <property name="owner" type="atom" description="SS type.
Available values: user | trunk | any" value="any"/>
            <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
                <item value="cfu"/>
                <item value="assistant"/>
                <item value="follow_me"/>
                <item value="tsmn_request"/>
            </property_list>
        </msd>
    </ss>

```

```

        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean" description=""
value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address" description="Forward
number"/>
        <property name="external_number" type="address"
description="External forward number" value=""/>
        <property name="use_external_number" type="boolean"
description="Use external forward number" value="false"/>
        <property name="timeout" type="positive_integer"
description="Timeout before call forward (sec)" value="15"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="sca" version="2" description="Selective Call
Acceptance" active="false" enabled="false" available="false">
    <csd>
        <property_list name="whitelist" type="index_address"
maxlen="50" description="White list members. Format: <address>[</index>]"/>
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="rbp" version="3" description="SS for restricting
call-processes by NI and PIN" active="false" enabled="false" available="false">
    <csd>
        <property name="implicit_call" type="boolean" description="Use
implicit RBP call (Request PIN-code in case of use recsricted call)" value="false"/>
        <property_list name="restricted_ni" type="ni" maxlen="6"
description="Restricted number indication">
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="cliro" version="2" description="Calling Line
Identification Restriction Override" active="false" enabled="false" available="false"/>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="intervention" version="4"
description="Intervention" active="false" enabled="false" available="false">
        <msd>

```



```

        <property name="enumRestriction" type="string"
description="Internal field to validate mode property" value="full_control|
implicitly_allowed|explicitly_allowed"/>
    </msd>
    <ssd>
        <property name="dtmf_detector" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="restriction" type="enum" description="The
restriction to the intervention call (full_control|allowed_only)" value="full_control"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="alarm" version="8" description="Alarm Call"
active="false" enabled="false" available="false">
    <msd>
        <property name="handle_module" type="atom" description=""
value="ss_alarm_handler"/>
    </msd>
    <ssd>
        <property name="collect_digits_timeout" type="integer"
description="Wait timeout to collect cancel digits" value="60"/>
    </ssd>
    <csd>
        <property name="no_answer_timeout" type="integer"
description="Awaiting answer timer (value in seconds)" value="45"/>
        <property name="call_attempt_count" type="integer"
description="The count of the call's attempt" value="3"/>
        <property name="call_attempt_timeout" type="integer"
description="Awaiting timer between calls' attempts" value="60"/>
        <property name="cancel_digits" type="address"
description="Digits to cancel alarm" value="*"/>
        <property name="alarm_1" type="term" description="{
'one-time' | daily, [" Day1[, Day2][, ...] "], {"HH, MM"}[, AlarmName]}" value="off"/>
        <property name="alarm_2" type="term" description="{
'one-time' | daily, [" Day1[, Day2][, ...] "], {"HH, MM"}[, AlarmName]}" value="off"/>
        <property name="alarm_3" type="term" description="{
'one-time' | daily, [" Day1[, Day2][, ...] "], {"HH, MM"}[, AlarmName]}" value="off"/>
        <property name="alarm_4" type="term" description="{
'one-time' | daily, [" Day1[, Day2][, ...] "], {"HH, MM"}[, AlarmName]}" value="off"/>
        <property name="alarm_5" type="term" description="{
'one-time' | daily, [" Day1[, Day2][, ...] "], {"HH, MM"}[, AlarmName]}" value="off"/>
        <property name="alarm_6" type="term" description="{
'one-time' | daily, [" Day1[, Day2][, ...] "], {"HH, MM"}[, AlarmName]}" value="off"/>
        <property name="alarm_7" type="term" description="{
'one-time' | daily, [" Day1[, Day2][, ...] "], {"HH, MM"}[, AlarmName]}" value="off"/>
        <property name="alarm_8" type="term" description="{
'one-time' | daily, [" Day1[, Day2][, ...] "], {"HH, MM"}[, AlarmName]}" value="off"/>

```

```

                <property name="alarm_9" type="term" description=""{"
''one-time'' | daily, "[" Day1[, Day2][, ...] "]" ,{"HH, MM"}"[, AlarmName]"}" value="off"/>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="speed_dial" version="5" description="Speed dial
(Abbreviated dialing)" active="false" enabled="false" available="false">
            <msd>
                <property name="enumType" type="string" description="The
type of address book. address_book_10 supports 10 number in the address book. address_book_100
supports 100 numbers on the address book" value="address_book_10|address_book_100"/>
            </msd>
            <csd>
                <property name="type" type="enum"
description="(address_book_10 | address_book_100)" value="address_book_100"/>
                <property_list name="map" type="speed_dial_item"
maxlen="50" description="Map between short dial and full address. Format: <short_dial>/
<address>"/>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="fax_to_email" version="2" description="Fax to
email" active="false" enabled="false" available="false">
            <csd>
                <property name="email" type="string" description="Email
address" value=""/>
                <property name="send_error_report" type="boolean"
description="Send email in case of fax received failed" value="false"/>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="flexicall" version="1" description="FlexiCall"
active="false" enabled="false" available="false">
            <msd>
                <property name="owner" type="atom" description="SS type.
Available values: user | trunk | any." value="user"/>
                <property name="enumMode" type="string"
description="Internal field to validate mode property" value="all|external|internal"/>
                <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
                    <item value="cgg"/>
                </property_list>
            </msd>
            <ssd>
                <property name="forwarding_ss" type="boolean"
description="" value="true"/>
            </ssd>
        </ss>

```

```

        <csd>
            <property name="mode" type="enum" description="Mode
operating. Available values (all | external | internal)"/>
            <property_list name="external_numbers" type="address"
maxlen="10" description="Group's external numbers"/>
            <property_list name="internal_numbers" type="address"
maxlen="10" description="Group's internal numbers"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="auto_redial_with_callback" version="5"
description="Auto redial with callback" active="false" enabled="false" available="false">
        <msd>
            <property name="enumRingback" type="string"
description="Ringback" value="tone|moh"/>
        </msd>
        <csd>
            <property name="recall_timer" type="positive_integer"
description="Time between recalls (sec)" value="90"/>
            <property name="recall_count" type="positive_integer"
description="Recalls' count" value="30"/>
            <property name="a_no_answer_timeout"
type="positive_integer" description="Awaiting B-side answer timeout (sec)" value="10"/>
            <property name="b_no_answer_timeout"
type="positive_integer" description="Awaiting A-side answer timeout (sec)" value="30"/>
            <property name="ringback" type="enum"
description="Ringback (tone | moh)" value="tone"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="park" version="2" description="Call park"
active="false" enabled="false" available="false"/>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="callback" version="5" description="CallBack"
active="false" enabled="false" available="false">
        <ssd>
            <property name="collect_digits_timeout" type="integer"
description="Wait timeout to collect cancel digits" value="60"/>
            <property name="a_onhook_timeout"
type="positive_integer" description="Timeout for A subscriber onhook (sec)" value="5"/>
            <property name="max_call_retry"
type="positive_integer" description="The maximum count of call's retry to initial subscriber"
value="5"/>
            <property name="dtmf_detector" type="boolean"
description="" value="true"/>
        </ssd>
    </ss>

```

```

        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="3way" version="6" description="3-Way Conference"
            active="false" enabled="false" available="false">
            <msd>
                <property_list name="dependency_list" type="atom"
                maxlen="0" description="List of SS that is depended from SS">
                    <item value="chold"/>
                </property_list>
                <property_list name="conflict_list" type="atom"
                maxlen="0" description="List of SS that is conflict with current SS">
                    <item value="conference"/>
                </property_list>
            </msd>
            <ssd>
                <property name="dtmf_detector" type="boolean"
                description="" value="true"/>
            </ssd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="introduce" version="1" description="Introduce
            announcement from callee" active="false" enabled="false" available="false">
            <csd>
                <property name="announcement" type="media_resource"
                description="Announcement media file"/>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="distinctive_picture" version="2"
            description="Display caller image by groups" active="false" enabled="false" available="false">
            <csd>
                <property_list name="groups"
                type="distinctive_picture" maxlen="20" description="Distinctive picture for specific numbers"/
                >
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="find_me_no_response" version="6"
            description="Find Me no response" active="false" enabled="false" available="false">
            <msd>
                <property_list name="conflict_list" type="atom"
                maxlen="0" description="List of SS that is conflict with current SS">
                    <item value="cfu"/>
                    <item value="assistant"/>
                    <item value="ctu"/>
                    <item value="cgg"/>
                    <item value="follow_me"/>
                </property_list>
            </msd>
        </ss>

```

```

                <item value="follow_me_no_response"/>
                <item value="find_me"/>
            </property_list>
        </msd>
        <ssd>
            <property name="forwarding_ss" type="boolean"
description="" value="false"/>
        </ssd>
        <csd>
            <property name="send_call_diversion" type="boolean"
description="Send call diversion flag in call signalisation" value="true"/>
            <property name="timeout" type="positive_integer"
description="Timeout before call forward (sec)" value="20"/>
            <property_list name="groups" type="find_me_group"
maxlen="32" description="Find me groups"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="follow_me_no_response" version="8"
description="Follow me no response" active="false" enabled="false" available="false">
        <msd>
            <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
                <item value="cfu"/>
                <item value="assistant"/>
                <item value="cft"/>
                <item value="cgg"/>
                <item value="follow_me"/>
            </property_list>
        </msd>
        <ssd>
            <property name="forwarding_ss" type="boolean"
description="" value="true"/>
        </ssd>
        <csd>
            <property name="number" type="address"
description="Followed number" value=""/>
            <property name="timeout" type="positive_integer"
description="Timeout before call forward (sec)" value="20"/>
            <property name="pin" type="address" description="PIN-
code" value=""/>
            <property name="external_number" type="address"
description="External forward number" value=""/>
            <property name="use_external_number" type="boolean"
description="Use external forward number" value="false"/>
        </csd>
    </ss>
</ss>

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="sco_white" version="2" description="Selective
        Call Originating, White list" active="false" enabled="false" available="false">
        <csd>
            <property_list name="whitelist" type="index_address"
            maxlen="50" description="White list members. Format: <address>[</index>]">
                <undefined/>
            </property_list>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="cc_agent" version="2" description="Contact-Center
        agent" active="false" enabled="false" available="false"/>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="tsmn_request" version="3" description="Support
            request on TSMN" active="false" enabled="false" available="false">
                <msd>
                    <property name="owner" type="atom" description="SS
                    type. Available values: user | trunk | any" value="trunk"/>
                </msd>
                <csd>
                    <property name="timeout" type="positive_integer"
                    description="Timeout on TSMN request(s) (msec)" value="3000"/>
                    <property name="tsmn_trunk" type="trunk"
                    description="Trunk to TSMN server"/>
                    <property name="tsmn_trunk_backup" type="trunk"
                    description="Backup trunk to TSMN server" value=""/>
                </csd>
            </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="dnd" version="3" description="Do not Disturb"
            active="false" enabled="false" available="false">
                <csd>
                    <property_list name="whitelist"
                    type="index_address" maxlen="10" description="White list members. Format: </
                    <index>]">
                        <undefined/>
                    </property_list>
                </csd>
            </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="second_handset" version="1" description="Second
            Handset" active="false" enabled="false" available="false"/>
        <ss

```

```

xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="rfc" version="2" description="Rejection
of Forwarded Calls" active="false" enabled="false" available="false"/>
  <ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="fbc" version="1" description="Forwarding
Barring Call" active="false" enabled="false" available="false">
    <msd>
      <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
    </msd>
  </ss>
  <ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfnr" version="2"
description="QSIG: Call Forwarding No Reply" active="false" enabled="false" available="false">
    <msd>
      <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="trunk"/>
      <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
        <item value="cfu"/>
      </property_list>
    </msd>
  </ss>
  <ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfb" version="11" description="Call
Forwarding Busy" active="false" enabled="false" available="false">
    <msd>
      <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
      <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
        <item value="cfu"/>
        <item value="assistant"/>
        <item value="follow_me"/>
        <item value="tsmn_request"/>
      </property_list>
    </msd>
    <ssd>
      <property name="forwarding_ss"
type="boolean" description="" value="true"/>
    </ssd>
    <csd>
      <property name="number" type="address"
description="Forward number"/>
      <property name="external_number"
type="address" description="External forward number" value=""/>

```

```

                <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="avatar" version="2" description="Picture
for outgoing calls" active="false" enabled="false" available="false">
            <csd>
                <property name="pic_addr"
type="uri_string" description="URL to access the image" value=""/>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfb" version="3" description="QSIG:
Call Forwarding Busy" active="false" enabled="false" available="false">
            <msd>
                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="trunk"/>
                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                    <item value="cfu"/>
                </property_list>
            </msd>
            <csd>
                <property name="hide_forwarding"
type="boolean" description="Hide forwarding fields (rgn, ocdrn)" value="false"/>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="fax_receiver" version="2"
description="Enable fax receiving ability for terminals which is not support fax by their own"
active="false" enabled="false" available="false"/>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="clir" version="3"
description="Calling Line Identification Restriction" active="false" enabled="false"
available="false">
            <msd>
                <property name="enumMode"
type="string" description="Internal field to validate mode property" value="user_defined|
restrict_all"/>
            </msd>
            <csd>
                <property name="mode" type="enum"
description="The mode of the working clir service (user_defined|restrict_all)"
value="restrict_all"/>
            </csd>
        </ss>
    </csd>

```



```

        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="teleconference_manager"
            version="3" description="Teleconference manager" active="false" enabled="false"
            available="false">
            <csd>
                <property_list name="second_line"
                    type="address" maxlen="10" description="Second line numbers"/>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="pickup" version="2"
            description="Call Pickup" active="false" enabled="false" available="false">
            <csd>
                <property_list name="pickup_groups"
                    type="index_string" maxlen="10" description="Call pick-up's groups names. Format: <Name>[/
                    <Inx>]">
                    <undefined/>
                </property_list>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="clip" version="5"
            description="Calling Line Identification Presentation" active="false" enabled="false"
            available="false"/>
        <ss
            xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cf_sip" version="9"
            description="Call Forwarding by SIP redirection (302 message)" active="false" enabled="false"
            available="false">
            <msd>
                <property name="owner" type="atom"
                    description="SS type. Available values: user | trunk | any" value="any"/>
                <property_list name="conflict_list
                    " type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                    <item value="cfu"/>
                    <item value="assistant"/>
                    <item value="follow_me"/>
                    <item value="tsmn_request"/>
                </property_list>
            </msd>
            <ssd>
                <property name="forwarding_ss"
                    type="boolean" description="" value="true"/>
            </ssd>
        </ss>
    </ss>

```

```

                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cpip" version="2"
description="Calling Picture Identification Presentation" active="false" enabled="false"
available="false">
                                <msd>
                                <property name="enumDisplay_mode"
type="string" description="Internal field to validate 'display-mode' property" value="icon|
fullscreen"/>
                                </msd>
                                <csd>
                                <property name="display_mode"
type="enum" description="Sets the picture display mode: icon mode or fullscreen mode."
value="icon"/>
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cnip" version="3"
description="Calling Name Identification Presentation" active="false" enabled="false"
available="false">
                                <msd>
                                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
                                </msd>
                                </ss>
                                </profile>

```

HC_ss_system_profile_list

Команда позволяет просмотреть список системных профилей услуг.

Метод HTTP-запроса:

Шаблон HTTP-запроса:

http://host:port/commands/ss_system_profile_list

Код HTTP-ответа:

200 – в случае успеха

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- profileType -->
  <xs:complexType name="profileType">
    <xs:attribute name="name" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          System's profile name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="description" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Human readable description of the current profile.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:attribute name="storage" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Data Storage cluster's name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="profile" type="profileType" minOccurs="0" maxOccurs="unbounded"
/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:schema>

```

Пример

Запрос:

http://192.168.1.21:9999/commands/ss_system_profile_list

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ss_system_profile_list.xsd">
  <request storage="ds1"/>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss_system_profile_list.xsd">
  <profile name="ss_test" description="test"/>
  <profile name="ss_biysk" description="ss_tstpfofile"/>
  <profile name="ss1" description="test"/>
</out>
```

HC_ss_system_profile_remove

Команда позволяет удалить несколько системных профилей по имени.

Аналог команды в CoCon:**c`l`uster/storage/<STORAGE>/ss/profile/remove****Метод HTTP-запроса:**

POST

Шаблон HTTP-запроса:http://host:port/commands/ss_system_profile_remove**Код HTTP-ответа:**

- 204 – в случае успеха;
- 200 – в случае ошибки (с указанием ошибки в теле ответа).

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- removeProfileType -->
  <xs:complexType name="removeProfileType">
    <xs:attribute name="name" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          System's profile name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="profile" type="removeProfileType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="storage" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Data Storage cluster's name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1

Запрос: http://192.168.1.21:9999/commands/ss_system_profile_remove

```
<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_system_profile_remove.xsd">
  <request storage="ds1">
    <profile name="ss_test"/>
  </request>
</in>
```

Ответ:

204

Пример 2

В случае, если профили с именами profile1, profile2, profile3 не существуют

Запрос: http://192.168.1.21:9999/commands/ss_system_profile_remove

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_system_profile_remove.xsd">
  <request storage="ds">
    <profile name="profile1"/>
    <profile name="profile2"/>
    <profile name="profile3"/>
  </request>
</in>
```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_system_profile_remove.xsd">
  <error cmd="hc_ss_system_profile_remove" reason="not_exists" body="not_exists"
  entity="profile1"/>
  <error cmd="hc_ss_system_profile_remove" reason="not_exists" body="not_exists"
  entity="profile2"/>
  <error cmd="hc_ss_system_profile_remove" reason="not_exists" body="not_exists"
  entity="profile3"/>
</out>
```

Нс_ss_system_profile_set

Команда позволяет создать новые и/или изменить существующие системные профили услуг.

Аналог команды в CoCon:

```
cluster/storage/<STORAGE>/ss/profile/declare
```

```
cluster/storage/<STORAGE>/ss/profile/ss-add
```

```
cluster/storage/<STORAGE>/ss/profile/ss-change
```

```
cluster/storage/<STORAGE>/ss/profile/ss-remove
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_system_profile_set

Код HTTP-ответа:

- 201 – в случае успеха;
- 200 – в случае ошибки (с указанием ошибки в теле ответа).

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- removeProfileType -->
  <xs:complexType name="removeProfileType">
    <xs:attribute name="name" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          System's profile name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="profile" type="removeProfileType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="storage" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Data Storage cluster's name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1

Запрос: http://192.168.1.21:9999/commands/ss_system_profile_set


```

<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_system_profile_set.xsd">
  <request storage="ds1">
    <profile name="profile1" description="description 1">
      <ss name="chold" enable="true" active="false"/>
      <ss name="cfu" enable="false" active="false"/>
      <ss name="cfnr" enable="true" active="true">
        <property value="5" key="timeout"/>
        <property value="235" key="number"/>
      </ss>
    </profile>
    <profile name="profile2" description="description 2">
      <ss name="chold" enable="true" active="false"/>
    </profile>
  </request>
</in>

```

Ответ:

201

Пример 2

В случае, если при изменения настроек профиля с именем profile3 произошла ошибка (при этом profile2 успешно изменен):

Запрос:

http://192.168.1.21:9999/commands/ss_system_profile_set

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_system_profile_set.xsd">
  <request storage="ds1">
    <profile name="profile3" description="description 1">
      <ss name="chold" enable="true" active="false"/>
      <ss name="cfu" enable="false" active="false"/>
      <ss name="cfnr" enable="true" active="true">
        <property value="5" key="timeout"/>
      </ss>
    </profile>
    <profile name="profile2" description="description 2">
      <ss name="chold" enable="true" active="false"/>
    </profile>
  </request>
</in>

```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_system_profile_set.xsd">
  <error cmd="hc_ss_system_profile_set" reason="undefined_properties"
  body="{undefined_properties, [{cfnr, [number]}]}" entity="profile3"/>
</out>
```

hc_ss_system_profile_add_ss

Команда позволяет добавить новый набор услуг в уже созданные системные профили.

Аналог команды в CoCon:

Шаблон http запроса:

http://host:port/commands/ss_system_profile_add_ss

Код http ответа:

201 – в случае успеха;

200 – в случае ошибки (с указанием ошибки в теле ответа).

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <!-- propertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          SS property's value, formatted by in the ss_activate command.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- ssType -->
  <xs:complexType name="ssType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="enable" type="xs:boolean" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
  </xs:complexType>
  <!-- setProfileType -->
  <xs:complexType name="setProfileType">
    <xs:sequence>
      <xs:element name="ss" type="ssType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          System's profile name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="profile" type="setProfileType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="storage" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Data Storage cluster's name.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  </xs:schema>

```

```

        </xs:attribute>
    </xs:complexType>
    <!-- in -->
    <xs:element name="in">
        <xs:complexType>
            <xs:all>
                <xs:element name="request" type="requestType"/>
            </xs:all>
        </xs:complexType>
    </xs:element>
    <!-- out -->
    <xs:element name="out">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/ss_system_profile_add_ss

```

<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_system_profile_add_ss.xsd">
  <request storage="ds1">
    <profile name="ss_test">
      <ss name="cfu" enable="false" active="false"/>
    </profile>
  </request>
</in>

```

Ответ:

201

Пример 2

В случае, если при изменении настроек профиля с именем profile3 произошла ошибка (при этом profile2 успешно изменен):

Запрос: http://192.168.1.21:9999/commands/ss_system_profile_add_ss

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ss_system_profile_add_ss.xsd">
  <request storage="ds1">
    <profile name="profile3">
      <ss name="chold" enable="true" active="false"/>
      <ss name="cfu" enable="false" active="false"/>
      <ss name="cfnr" enable="true" active="true">
        <property value="5" key="timeout"/>
      </ss>
    </profile>
    <profile name="profile2">
      <ss name="chold" enable="true" active="false"/>
    </profile>
  </request>
</in>
```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss_system_profile_add_ss.xsd">
  <error cmd="hc_ss_system_profile_add_ss" reason="undefined_properties"
body="{undefined_properties, [{cfnr, [number]}}" entity="profile3"/>
</out>
```

Нс_ss_trunk - Команды управления транковыми сервисами

- [Нс_ss_trunk_activate](#)
- [Нс_ss_trunk_deactivate](#)
- [Нс_ss_trunk_info](#)

Нс_ss_trunk_activate

Команда предназначена для активации сервиса на определенном транке в определенном домене.

Аналог команды в CoCon:

domain/<DOMAIN>/trunk/services/activate

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:9999/commands/ss_trunk_activate

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="trunk" type="xs:string" use="optional"/>
    <xs:attribute name="ss" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="trunk" type="xs:string" use="optional"/>
    <xs:attribute name="ss" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- activateSsPropertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- ssActivateType -->
  <xs:complexType name="ssActivateType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="trunk" type="xs:string" use="required"/>
    <xs:attribute name="ss" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="activate" type="ssActivateType"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>

```

```

        <xs:element name="ok" type="okType" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="error" type="errorType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
</xs:complexType>
</xs:element>
</xs:all>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1:

Запрос:

http://192.168.1.21:9999/commands/ss_trunk_activate

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ss_trunk_activate.xsd">
  <request domain="biysk.local">
    <activate ss="cfu" trunk="ems2">
      <property value="240469" key="number"/>
    </activate>
    <activate ss="cfnr" trunk="bsk1">
      <property value="240464" key="number"/>
      <property value="5" key="timeout"/>
    </activate>
  </request>
</in>

```

Ответ:

Код 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_trunk_activate.xsd">
  <response domain="biysk.local">
    <error reason="{undefined_properties,[mode]}" trunk="smg-4" ss="call_recording"/>
  </response>
</out>

```

Пример 2:

Запрос:

http://192.168.1.21:9999/commands/ss_trunk_activate

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_trunk_activate.xsd">
  <request domain="biysk.local">
    <activate ss="cfnr" trunk="ems1">
      <property value="14312" key="number"/>
      <property value="5" key="timeout"/>
    </activate>
  </request>
</in>
```

Ответ:

Код 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_trunk_activate.xsd">
  <response domain="biysk.local">
    <ok trunk="ems1" ss="cfnr"/>
  </response>
</out>
```

[Hc_ss_trunk_deactivate](#)

Команда предназначена для деактивации транковых сервисов в определенно домене.

Аналог команды в CoCon:

domain/<DOMAIN>/trunk/services/deactivate

Метод HTTP-запроса

POST

Шаблон HTTP-запроса:

http://host:9999/commands/ss_trunk_deactivate

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом/ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="trunk" type="xs:string" use="optional"/>
    <xs:attribute name="ss" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="trunk" type="xs:string" use="optional"/>
    <xs:attribute name="ss" type="xs:string" use="optional"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- ssDeactivateType -->
  <xs:complexType name="ssDeactivateType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="trunk" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ss" type="ssDeactivateType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok" type="okType" minOccurs="0"
maxOccurs="unbounded"/>
              <xs:element name="error" type="errorType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

        </xs:element>
    </xs:all>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Деактивировать услугу **cfu**, **cfnr** у транков **ems2**, **bsk1** в домене **biysk.local**.

Запрос:

http://192.168.34.87:9999/commands/ss_trunk_deactivate

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_trunk_deactivate.xsd">
  <request domain="biysk.local">
    <ss trunk="ems1" name="cfnr" />
    <ss trunk="smg-4" name="call_recording" />
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_trunk_deactivate.xsd">
  <response domain="biysk.local">
    <ok trunk="smg-4" ss="call_recording"/>
    <ok trunk="ems1" ss="cfnr"/>
  </response>
</out>

```

[Hc_ss_trunk_info](#)

Команда возвращает список активированных транковых сервисов в рамках определенного домена.

Аналог команды в CoCon:

domain/<DOMAIN>/trunk/services/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/ss_trunk_info`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

ss.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- Migration start -->
  <!-- ruleActionType -->
  <xs:simpleType name="ruleActionType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="default"/>
      <xs:enumeration value="copy"/>
      <xs:enumeration value="set"/>
      <xs:enumeration value="custom_migrator"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- ruleType -->
  <xs:complexType name="ruleType">
    <xs:attribute name="action" type="ruleActionType" use="required"/>
    <xs:attribute name="property" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- migrationType -->
  <xs:complexType name="migrationType">
    <xs:sequence>
      <xs:element name="rule" type="ruleType" minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="version" type="xs:integer" use="required"/>
  </xs:complexType>
  <!-- Migration end -->
  <!-- propertyTypeType -->
  <xs:simpleType name="propertyTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="atom"/>
      <xs:enumeration value="enum"/>
      <xs:enumeration value="regime"/>
      <xs:enumeration value="boolean"/>
      <xs:enumeration value="address"/>
      <xs:enumeration value="interface"/>
      <xs:enumeration value="trunk"/>
      <xs:enumeration value="domain"/>
      <xs:enumeration value="hour"/>
      <xs:enumeration value="minute"/>
      <xs:enumeration value="second"/>
      <xs:enumeration value="string"/>
      <xs:enumeration value="ni"/>
      <xs:enumeration value="index_string"/>
      <xs:enumeration value="index_address"/>
      <xs:enumeration value="integer"/>
      <xs:enumeration value="float"/>
      <xs:enumeration value="positive_integer"/>
      <xs:enumeration value="positive_float"/>
      <xs:enumeration value="negative_integer"/>
    </xs:restriction>
  </xs:simpleType>

```

```

<xs:enumeration value="negative_float"/>
<xs:enumeration value="term"/>
<xs:enumeration value="ip"/>
<xs:enumeration value="integer_range"/>
<xs:enumeration value="speed_dial_item">
  <xs:annotation>
    <xs:documentation xml:lang="ru">
      speed_dial_item представляет собой тапл: {КороткийНабор, ПолныйАдрес}
      В xml данное поле представлено в виде КороткийНабор/ПолныйАдрес
    </xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="ivr_script_id">
  <xs:annotation>
    <xs:documentation xml:lang="ru">
      Идентификатор IVR скрипта в домене
    </xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="teleconference_room">
  <xs:annotation>
    <xs:documentation xml:lang="ru">
      Номер комнаты телеконференции в домене
    </xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="media_resource">
  <xs:annotation>
    <xs:documentation xml:lang="ru">
      media_resource представляет собой строку формата: {system|domain|tone}://
      PATH_OR_TONE
      Sample 1: domain://sounds/rus/alarm.wav
      Sample 2: system://sounds/ai_agent_already_on.wav
      Sample 3: tone://?f2=425&off_msec=4000&on_msec=1000&f1=0&duration=300000
    </xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="find_me_group">
  <xs:annotation>
    <xs:documentation xml:lang="ru">
      find_me_group является сложным типом данных, используется только в ДВО
      FindMe, FindMe not response.
      В xml данное поле записывается в виде JSON структуры данных, имеющее вид:
      {"numbers":"NUMBERS_LIST","wait_timeout":"WAIT_TIMEOUT","enabled":"ENABLE
      D","work_interval":WORK_INTERVAL},
      NUMBERS_LIST - список номеров группы, записанных через запятую;
      WAIT_TIMEOUT - время ожидания ответа участника группы (5..90 секунд);
      ENABLED :: true | false - активна или нет данная группа;
      WORK_INTERVAL :: {"type":"predefined", "timetable":"TIMETABLE"} |

```

```

        {"type":"custom", "days":"DAYS", "times":[{"from":"FROM_TIME","to":"TO_TIME"}, {"from":"FROM_TIME", "to":"TO_TIME"}, ...]}
    в случае, если type = predefined, то указывается только параметр
timetable - имя предопределенного интервала времени;
    в случае, если type = custom, то необходимо указать параметры days, times
(times - список интервалов).
    DAYS ::DAY_1DAY_2... - список дней недели (1..7) которые входят в данный
интервал (записываются слитно);
    FROM_TIME :: FROM_HH:FROM_MM - начало интервала времени;
    TO_TIME :: TO_HH:TO_MM - конец интервала времени.
    Например 1: {"numbers":"710,711","wait_timeout":60,"enabled":true,"work_interval":{"type":"predefined","timetable":"work_time"}}
    Например 2: {"numbers":"712","wait_timeout":90,"enabled":false,"work_interval":{"type":"custom", "days":"67", "times":[{"from":"0:0", "to":"17:59"}, {"from":"18:0", "to":"23:59"}]}}
    </xs:documentation>
    </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="version"/>
    </xs:restriction>
</xs:simpleType>
<!-- propertyType -->
<xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="propertyTypeType" use="required"/>
    <xs:attribute name="value" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
</xs:complexType>
<!-- propertyListType -->
<xs:complexType name="propertyListType">
    <xs:choice>
        <xs:sequence>
            <xs:element name="item" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                    <xs:attribute name="value" type="xs:string" use="required"/>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
        <xs:element name="undefined" minOccurs="0">
            <xs:complexType/>
        </xs:element>
    </xs:choice>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="propertyTypeType" use="required"/>
    <xs:attribute name="maxlen" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
</xs:complexType>
<!-- ssType -->
<xs:complexType name="ssType">
    <xs:sequence>

```



```

<xs:element name="migrations" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="migration" type="migrationType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="msd" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded"/>
      <xs:element name="property_list" type="propertyListType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ssd" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded"/>
      <xs:element name="property_list" type="propertyListType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="csd" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0"
maxOccurs="unbounded"/>
      <xs:element name="property_list" type="propertyListType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="name" type="xs:string" use="required"/>
<xs:attribute name="family" type="xs:string" use="optional">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Obsolete attribute. Use name instead of It.
    </xs:documentation>
    <xs:documentation xml:lang="ru">
      Устаревший атрибут. Теперь ДВО полностью определяется по атрибуту name.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>

```

```

    <xs:attribute name="version" type="xs:integer" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="enabled" type="xs:boolean" use="optional"/>
    <xs:attribute name="active" type="xs:boolean" use="optional"/>
  </xs:complexType>
  <!-- activatedSsType -->
  <xs:complexType name="activatedSsType">
    <xs:complexContent>
      <xs:extension base="ssType">
        <xs:attribute name="domain" type="xs:string" use="required"/>
        <xs:attribute name="address" type="xs:string" use="required"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- rawSsType -->
  <xs:complexType name="srcSsType">
    <xs:complexContent>
      <xs:extension base="ssType">
        <xs:attribute name="src" type="xs:string" use="required"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- ssListType -->
  <xs:complexType name="ssListType">
    <xs:choice>
      <xs:sequence>
        <xs:element name="ss" type="ssType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:sequence>
        <xs:element name="activated_ss" type="activatedSsType" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:sequence>
        <xs:element name="src_ss" type="srcSsType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:choice>
  </xs:complexType>
  <!-- ss -->
  <xs:element name="ss" type="ssType"/>
</xs:schema>

```

ss_trunk_info.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ss.xsd"/>

  <xs:complexType name="ssInfoType">
    <xs:complexContent>
      <xs:extension base="ssType">
        <xs:attribute name="domain" type="xs:string" use="required"/>
        <xs:attribute name="trunk" type="xs:string" use="required"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- ssListType -->
  <xs:complexType name="ssInfoListType">
    <xs:sequence>
      <xs:element name="ss" type="ssInfoType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="trunk" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="sss" type="ssInfoListType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/ss_trunk_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ss_trunk_info.xsd">
  <request domain="biysk.local" trunk="ems2"/>
</in>
```

Ответ:200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_trunk_info.xsd">
  <sss>
    <ss domain="biysk.local" trunk="bsk1"
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="tsmn_request" version="3" description="Support
      request on TSMN" active="true" enabled="false">
      <msd>
        <property name="owner" type="atom" description="SS type. Available values: user |
        trunk | any" value="trunk"/>
      </msd>
      <csd>
        <property name="timeout" type="positive_integer" description="Timeout on TSMN
        request(s) (msec)" value="3"/>
        <property name="tsmn_trunk" type="trunk" description="Trunk to TSMN server"
        value="smg-4"/>
        <property name="tsmn_trunk_backup" type="trunk" description="Backup trunk to TSMN
        server" value="nsk_sbc"/>
      </csd>
    </ss>
    <ss domain="biysk.local" trunk="bsk1"
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="call_recording" version="10" description="Call
      recording" active="false" enabled="false">
      <msd>
        <property name="enumMode" type="string" description="Internal field to validate
        mode property" value="always_on|on_demand|after_answer"/>
        <property name="owner" type="atom" description="SS type. Available values: user |
        trunk | any" value="any"/>
      </msd>
      <ssd>
        <property name="dtmf_detector" type="boolean" description="" value="true"/>
        <property name="locked" type="boolean" description="Can subscriber manage current
        SS, or not" value="true"/>
      </ssd>
      <csd>
        <property name="mode" type="enum" description="The mode of the recording call
        (always_on|on_demand)" value="on_demand"/>
      </csd>
    </ss>
    <ss domain="biysk.local" trunk="bsk1"
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="fxo_trunk" version="1" description="FXO trunk
      service" active="false" enabled="false">
      <msd>
        <property name="owner" type="atom" description="SS type. Available values: user |
        trunk | any" value="trunk"/>

```

```

    </msd>
  </ss>
  <ss domain="biysk.local" trunk="bsk1"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="message" version="2" description="Message service"
    active="false" enabled="false">
    <msd>
      <property name="owner" type="atom" description="SS type. Available values: user |
trunk | any" value="any"/>
    </msd>
    <ssd>
      <property name="message_box_size" type="integer" description="Message box size"
value="20"/>
      <property name="remove_only_after_answer" type="boolean" description="Remove
message only after abonent answer" value="false"/>
    </ssd>
  </ss>
  <ss domain="biysk.local" trunk="bsk1"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="cfos" version="12" description="Call Forwarding Out
of Service" active="false" enabled="false">
    <msd>
      <property name="owner" type="atom" description="SS type. Available values: user |
trunk | any" value="any"/>
      <property_list name="conflict_list" type="atom" maxlen="0" description="List of
SS that is conflict with current SS">
        <item value="cfu"/>
        <item value="assistant"/>
        <item value="follow_me"/>
        <item value="tsmn_request"/>
      </property_list>
    </msd>
    <ssd>
      <property name="forwarding_ss" type="boolean" description="" value="true"/>
    </ssd>
    <csd>
      <property name="number" type="address" description="Forward number"/>
      <property name="external_number" type="address" description="External forward
number" value=""/>
      <property name="use_external_number" type="boolean" description="Use external
forward number" value="false"/>
    </csd>
  </ss>
  <ss domain="biysk.local" trunk="bsk1"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="cf_aon" version="8" description="Call Forwarding AON"
    active="false" enabled="false">
    <msd>
      <property name="owner" type="atom" description="SS type. Available values: user |
trunk | any" value="any"/>

```

```

        <property_list name="conflict_list" type="atom" maxlen="0" description="List of
SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
            <item value="cfu"/>
            <item value="find_me"/>
            <item value="cf_aon"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="no_reply" type="boolean" description="Activate service when
subscriber no reply" value="false"/>
        <property name="timeout" type="positive_integer" description="Timeout before call
forward (sec)" value="15"/>
        <property name="number" type="address" description="Forward number"/>
        <property_list name="prefix_number" type="address" maxlen="10"
description="Prefix number A">
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss domain="biysk.local" trunk="bsk1"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cfu" version="14" description="Call Forwarding
Unconditional" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available values: user |
trunk | any" value="any"/>
        <property_list name="conflict_list" type="atom" maxlen="0" description="List of
SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address" description="Forward number"/>
        <property name="external_number" type="address" description="External forward
number" value=""/>
        <property name="use_external_number" type="boolean" description="Use external
forward number" value="false"/>

```

```

    </csd>
  </ss>
  <ss domain="biysk.local" trunk="bsk1"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfu" version="3" description="QSIG: Call
    Forwarding Unconditional" active="false" enabled="false">
    <msd>
      <property name="owner" type="atom" description="SS type. Available values: user |
      trunk | any" value="trunk"/>
      <property_list name="conflict_list" type="atom" maxlen="0" description="List of
      SS that is conflict with current SS">
        <item value="cfu"/>
        <item value="follow_me"/>
      </property_list>
    </msd>
  </csd>
  <property name="hide_forwarding" type="boolean" description="Hide forwarding
  fields (rgn, ocdrn)" value="false"/>
</csd>
</ss>
<ss domain="biysk.local" trunk="bsk1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="cft" version="14" description="Call Forwarding
  Unconditional Time Dependent" active="false" enabled="false">
  <msd>
    <property name="owner" type="atom" description="SS type. Available values: user |
    trunk | any" value="any"/>
    <property name="enumTime_format" type="string" description="Internal field to
    validate time_format property" value="local|utc"/>
    <property_list name="conflict_list" type="atom" maxlen="0" description="List of
    SS that is conflict with current SS">
      <item value="assistant"/>
      <item value="follow_me"/>
      <item value="cfu_type2"/>
      <item value="meet_me_conference"/>
      <item value="cfu"/>
      <item value="find_me"/>
      <item value="cf_aon"/>
    </property_list>
  </msd>
  <ssd>
    <property name="forwarding_ss" type="boolean" description="" value="true"/>
  </ssd>
  <csd>
    <property name="day_1" type="term" description="{{{FromHour,FromMinute},
    {ToHour,ToMinute}}, ..., Number, ExternalNumber}" value="off"/>
    <property name="day_2" type="term" description="{{{FromHour,FromMinute},
    {ToHour,ToMinute}}, ..., Number, ExternalNumber}" value="off"/>
    <property name="day_3" type="term" description="{{{FromHour,FromMinute},
    {ToHour,ToMinute}}, ..., Number, ExternalNumber}" value="off"/>
  </csd>
</ss>

```



```

        <property name="day_4" type="term" description="{{{FromHour,FromMinute},
{ToHour,ToMinute}}, ...], Number, ExternalNumber}" value="off"/>
        <property name="day_5" type="term" description="{{{FromHour,FromMinute},
{ToHour,ToMinute}}, ...], Number, ExternalNumber}" value="off"/>
        <property name="day_6" type="term" description="{{{FromHour,FromMinute},
{ToHour,ToMinute}}, ...], Number, ExternalNumber}" value="off"/>
        <property name="day_7" type="term" description="{{{FromHour,FromMinute},
{ToHour,ToMinute}}, ...], Number, ExternalNumber}" value="off"/>
        <property name="time_format" type="enum" description="CFT for Sunday (local |
utc)" value="local"/>
    </csd>
</ss>
<ss domain="biysk.local" trunk="bsk1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="cfnr_type2" version="11" description="Call Forwarding
No Reply without notification" active="false" enabled="false">
  <msd>
    <property name="owner" type="atom" description="SS type. Available values: user |
trunk | any" value="any"/>
    <property_list name="conflict_list" type="atom" maxlen="0" description="List of
SS that is conflict with current SS">
      <item value="cfu"/>
      <item value="assistant"/>
      <item value="cfnr"/>
      <item value="follow_me"/>
      <item value="tsmn_request"/>
    </property_list>
  </msd>
  <ssd>
    <property name="forwarding_ss" type="boolean" description="" value="true"/>
  </ssd>
  <csd>
    <property name="number" type="address" description="Forward number"/>
    <property name="timeout" type="positive_integer" description="Timeout before call
forward (sec)" value="15"/>
    <property name="use_redirection_counter" type="boolean" description="Use
redirectionCounter field to count hops" value="false"/>
    <property name="external_number" type="address" description="External forward
number" value=""/>
    <property name="use_external_number" type="boolean" description="Use external
forward number" value="false"/>
  </csd>
</ss>
<ss domain="biysk.local" trunk="bsk1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="cfu_type2" version="14" description="Call Forwarding
Unconditional without notification" active="false" enabled="false">
  <msd>
    <property name="owner" type="atom" description="SS type. Available values: user |
trunk | any" value="any"/>

```

```

        <property_list name="conflict_list" type="atom" maxlen="0" description="List of
SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
        </property_list>
    </msd>
</ssd>
    <property name="forwarding_ss" type="boolean" description="" value="true"/>
</ssd>
<csd>
    <property name="number" type="address" description="Forward number"/>
    <property name="use_redirection_counter" type="boolean" description="Use
redirectionCounter field to count hops" value="false"/>
    <property name="external_number" type="address" description="External forward
number" value=""/>
    <property name="use_external_number" type="boolean" description="Use external
forward number" value="false"/>
</csd>
</ss>
<ss domain="biysk.local" trunk="bsk1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="cfnr" version="11" description="Call Forwarding No
Reply" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available values: user |
trunk | any" value="any"/>
        <property_list name="conflict_list" type="atom" maxlen="0" description="List of
SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="tsmn_request"/>
        </property_list>
    </msd>
</ssd>
    <property name="forwarding_ss" type="boolean" description="" value="true"/>
</ssd>
<csd>
    <property name="number" type="address" description="Forward number"/>
    <property name="external_number" type="address" description="External forward
number" value=""/>
    <property name="use_external_number" type="boolean" description="Use external
forward number" value="false"/>
    <property name="timeout" type="positive_integer" description="Timeout before call
forward (sec)" value="15"/>
</csd>
</ss>
<ss domain="biysk.local" trunk="bsk1"

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="fbc" version="1" description="Forwarding Barring
Call" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available values: user |
trunk | any" value="any"/>
    </msd>
</ss>
<ss domain="biysk.local" trunk="bsk1"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfnr" version="2" description="QSIG: Call
Forwarding No Reply" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available values: user |
trunk | any" value="trunk"/>
        <property_list name="conflict_list" type="atom" maxlen="0" description="List of
SS that is conflict with current SS">
            <item value="cfu"/>
        </property_list>
    </msd>
</ss>
<ss domain="biysk.local" trunk="bsk1"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cfb" version="11" description="Call Forwarding Busy"
active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available values: user |
trunk | any" value="any"/>
        <property_list name="conflict_list" type="atom" maxlen="0" description="List of
SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="tsmn_request"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address" description="Forward number"/>
        <property name="external_number" type="address" description="External forward
number" value=""/>
        <property name="use_external_number" type="boolean" description="Use external
forward number" value="false"/>
    </csd>
</ss>
<ss domain="biysk.local" trunk="bsk1"

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfb" version="3" description="QSIG: Call
Forwarding Busy" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available values: user |
trunk | any" value="trunk"/>
        <property_list name="conflict_list" type="atom" maxlen="0" description="List of
SS that is conflict with current SS">
            <item value="cfu"/>
        </property_list>
    </msd>
    <csd>
        <property name="hide_forwarding" type="boolean" description="Hide forwarding
fields (rgn, ocdrn)" value="false"/>
    </csd>
</ss>
<ss domain="biysk.local" trunk="bsk1"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cf_sip" version="9" description="Call Forwarding by
SIP redirection (302 message)" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available values: user |
trunk | any" value="any"/>
        <property_list name="conflict_list" type="atom" maxlen="0" description="List of
SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="tsmn_request"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean" description="" value="true"/>
    </ssd>
</ss>
<ss domain="biysk.local" trunk="bsk1"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="cnip" version="3" description="Calling Name
Identification Presentation" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available values: user |
trunk | any" value="any"/>
    </msd>
</ss>
</sss>
</out>

```

Нс_ss_voicemail - Команды управления сервисом голосовой почты

- [Hc_ss_voicemail_mailbox_modify](#)
- [Hc_ss_voicemail_mailbox_show](#)

Hc_ss_voicemail_mailbox_modify

Изменить сообщения голосовой почты.

Команда позволяет промаркировать определенные сообщения голосовой почты.

Варианты маркировки :

- save - сохранить выбранный файл
- delete - удалить выбранный файл
- mark-as-read - пометить файл прочитанным
- mark-as-unread - пометить файл не прочитанным

Аналог команды в CoCon:

`domain/<DOMAIN>/ss/voicemail/mailbox/modify`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/ss_voicemail_mailbox_modify`

Код http ответа

204 – в случае успеха;

404 – в случае ошибки (с указанием ошибки в теле ответа).

Xsd схема XML запроса/ответа

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- modifyType -->
  <xs:simpleType name="modifyType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="mark-as-read"/>
      <xs:enumeration value="mark-as-unread"/>
      <xs:enumeration value="save"/>
      <xs:enumeration value="delete"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- modifyRecordType-->
  <xs:complexType name="modifyRecordType">
    <xs:attribute name="action" type="modifyType" use="required"/>
    <xs:attribute name="url" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- requestType-->
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="modify" type="modifyRecordType" minOccurs="0" maxOccurs="unbounded"
/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1

Запрос на изменение существующей записи: http://192.168.1.21:9999/commands/ss_voicemail_mailbox_modify. При этом файл перемещается в папку с именем save/.

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_voicemail_mailbox_modify.xsd">
  <request domain="biysk.local" address="240473">
    <modify action = "save" url = "http://system.restfs.ecss:9990/domain/biysk.local/
  subscribers/240473/voicemail/new/2019-11-15-09-53-32-83854416370.wav"/>
  </request>
</in>
```

Ответ: 204

Пример 2

Запрос на изменение НЕ существующей записи (записи после изменения): http://192.168.1.21:9999/commands/ss_voicemail_mailbox_modify Файла в папке new уже нет.

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_voicemail_mailbox_modify.xsd">
  <request domain="biysk.local" address="240473">
    <modify action = "save" url = "http://system.restfs.ecss:9990/domain/biysk.local/
  subscribers/240473/voicemail/new/2019-11-15-09-53-32-83854416370.wav"/>
  </request>
</in>
```

Ответ: 404

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_voicemail_mailbox_modify.xsd">
  <error cmd="hc_ss_voicemail_mailbox_modify" reason="nothing_to_modify"
  body="{error,nothing_to_modify}"/>
</out>
```

Пример 3

Запрос на изменение записи по не существующему адресу: http://192.168.1.21:9999/commands/ss_voicemail_mailbox_modify

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_voicemail_mailbox_modify.xsd">
  <request domain="biysk.local" address="240999">
    <modify action = "save" url = "http://system.restfs.ecss:9990/domain/biysk.local/
  subscribers/240464/voicemail/new/2017-12-10-10-49-31-240471.wav"/>
  </request>
</in>
```

Ответ: 404

```
<?xml version="1.0" encoding="UTF-8"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_voicemail_mailbox_modify.xsd">
  <error cmd="hc_ss_voicemail_mailbox_modify" reason="wrong_address"
  body="{error,wrong_address}"/>
</out>
```

Пример: 4

Пометить запись голосовой почты как прочитанную.

Запрос:

```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="ss_voice
  mail_mailbox_modify.xsd">
  <request domain="biysk.local" address="240473">
    <modify action = "mark-as-read" url = "http://system.restfs.ecss:9990/domain/biysk.local/
  subscribers/240473/voicemail/new/2019-11-15-09-53-32-83854416370.wav"/>
  </request>
</in>
```


Ответ: 204

Пример: 5

Удалить запись голосовой почты .


```
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="ss_voice
mail_mailbox_modify.xsd">
  <request domain="biysk.local" address="240473">
    <modify action = "delete" url = "http://system.restfs.ecss:9990/domain/biysk.local/
subscribers/240473/voicemail/save/2019-11-15-09-53-32-83854416370.wav"/>
  </request>
</in>
```

Ответ: 204

 Начиная с версии 3.14.4 теперь можно использовать вместо указания URL символ * для удаления всех записей в vmail box.

Нс_ss_voicemail_mailbox_show

Просмотреть сообщения голосовой почты.

Аналог команды в CoCon:

domain/<DOMAIN>/ss/voicemail/mailbox/show

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_voicemail_mailbox_show

Код http ответа

204 – в случае успеха;

404 – в случае ошибки (с указанием ошибки в теле ответа).

Xsd схема XML запроса/ответа

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- recordType-->
  <xs:complexType name="recordType">
    <xs:attribute name="from" type="xs:string" use="required"/>
    <xs:attribute name="time" type="xs:string" use="required"/>
    <xs:attribute name="duration" type="xs:integer" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The duration of the calls' record (sec).
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="url" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- requestType-->
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- responseType-->
  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="unread" minOccurs="0" maxOccurs="unbounded" type="recordType"/>
      <xs:element name="read" minOccurs="0" maxOccurs="unbounded" type="recordType"/>
      <xs:element name="save" minOccurs="0" maxOccurs="unbounded" type="recordType"/>
    </xs:sequence>
    <xs:attribute name="size" type="xs:integer" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The size of the mailbox (sec).
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос на просмотр списка голосовых сообщений:

http://192.168.1.21:9999/commands/ss_voicemail_mailbox_show

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="ss_voicemail_mailbox_show.xsd">
  <response size="6">
    <unread url="http://system.restfs.ecss:9990/domain/biysk.local/subscribers/240473/voicemail/new/2019-11-15-09-53-32-83854416370.wav" duration="6" time="15.11.2019 02:53:32" from="83854416370"/>
  </response>
</out>

```

Ответ:200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="ss_voicemail_mailbox_show.xsd">
  <response size="6">
    <unread url="http://system.restfs.ecss:9990/domain/biysk.local/subscribers/240473/voicemail/new/2019-11-15-09-53-32-83854416370.wav" duration="6" time="15.11.2019 02:53:32" from="83854416370"/>
  </response>
</out>

```

Нс_ss - Управление услугами

- [Общая схема управления услугами](#)
- [Нс_ss_acl_add](#)
- [Нс_ss_acl_remove](#)
- [Нс_ss_acl_show](#)
- [Нс_ss_activate](#)
- [Нс_ss_available_show](#)
- [Нс_ss_calls_history](#)
- [Нс_ss_deactivate](#)
- [Нс_ss_enable](#)
- [Нс_ss_enabled](#)

- [Hc_ss_info](#)
- [Hc_ss_install](#)
- [Hc_ss_show](#)
- [Hc_ss_uninstall](#)

Общая схема управления услугами

ss.xsd:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">
  <!-- Migration start -->
  <!-- ruleActionType -->
  <xs:simpleType name="ruleActionType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="default"/>
      <xs:enumeration value="copy"/>
      <xs:enumeration value="set"/>
      <xs:enumeration value="remove"/>
      <xs:enumeration value="custom_migrator"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- ruleType -->
  <xs:complexType name="ruleType">
    <xs:attribute name="action" type="ruleActionType" use="required"/>
    <xs:attribute name="property" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- migrationType -->
  <xs:complexType name="migrationType">
    <xs:sequence>
      <xs:element name="rule" type="ruleType" minOccurs="0"
maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="version" type="xs:token" use="required"/>
  </xs:complexType>
  <!-- Migration end -->
  <!-- propertyTypeType -->
  <xs:simpleType name="propertyTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="atom"/>
      <xs:enumeration value="enum"/>
      <xs:enumeration value="regime"/>
      <xs:enumeration value="boolean"/>
      <xs:enumeration value="address"/>
      <xs:enumeration value="interface"/>
    </xs:restriction>
  </xs:simpleType>
</xs:schema>
```

```

<xs:enumeration value="trunk"/>
<xs:enumeration value="domain"/>
<xs:enumeration value="hour"/>
<xs:enumeration value="minute"/>
<xs:enumeration value="second"/>
<xs:enumeration value="string"/>
<xs:enumeration value="ni"/>
<xs:enumeration value="index_string"/>
<xs:enumeration value="index_address"/>
<xs:enumeration value="integer"/>
<xs:enumeration value="float"/>
<xs:enumeration value="positive_integer"/>
<xs:enumeration value="positive_float"/>
<xs:enumeration value="negative_integer"/>
<xs:enumeration value="negative_float"/>
<xs:enumeration value="term"/>
<xs:enumeration value="ip"/>
<xs:enumeration value="integer_range"/>
<xs:enumeration value="speed_dial_item">
  <xs:annotation>
    <xs:documentation xml:lang="ru">
      speed_dial_item представляет собой тапл:
      {КороткийНабор, ПолныйАдрес}
      В xml данное поле представлено в виде КороткийНабор/
      ПолныйАдрес
    </xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="ivr_script_id">
  <xs:annotation>
    <xs:documentation xml:lang="ru">
      Идентификатор IVR скрипта в домене
    </xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="media_resource">
  <xs:annotation>
    <xs:documentation xml:lang="ru">
      media_resource представляет собой строку формата:
      {system|domain|tone}://PATH_OR_TONE
      Sample 1: domain://sounds/rus/alarm.wav
      Sample 2: system://sounds/ai_agent_already_on.wav
      Sample 3: tone://?
      f2=425&off_msec=4000&on_msec=1000&f1=0&duration=300000
    </xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="find_me_group">

```

```

        <xs:annotation>
          <xs:documentation xml:lang="ru">
            find_me_group является сложным типом данных,
используется только в ДВО FindMe, FindMe not response.
            В xml данное поле записывается в виде JSON структуры
данных, имеющее вид:

{"numbers":"NUMBERS_LIST","wait_timeout":"WAIT_TIMEOUT","enabled":"ENABLED","
work_interval":WORK_INTERVAL},
            NUMBERS_LIST - список номеров группы, записанных
через запятую;
            WAIT_TIMEOUT - время ожидания ответа участника группы
(5..90 секунд);
            ENABLED :: true | false - активна или нет данная
группа;
            WORK_INTERVAL :: {"type":"predefined",
"timetable":"TIMETABLE"} |
            {"type":"custom", "days":"DAYS", "times":
[{"from":"FROM_TIME","to":"TO_TIME"},
{"from":"FROM_TIME","to":"TO_TIME"},...]}
            в случае, если type = predefined, то указывается
только параметр timetable - имя предопределенного интервала времени;
            в случае, если type = custom, то необходимо указать
параметры days, times (times - список интервалов).
            DAYS :: DAY_1DAY_2... - список дней недели (1..7)
которые входят в данный интервал (записываются слитно);
            FROM_TIME :: FROM_HH:FROM_MM - начало интервала
времени;
            TO_TIME :: TO_HH:TO_MM - конец интервала времени.
            Например 1: {"numbers":"710,711","wait_timeout":
60,"enabled":true,"work_interval":
{"type":"predefined","timetable":"work_time"}}
            Например 2: {"numbers":"712","wait_timeout":
90,"enabled":false,"work_interval":{"type":"custom", "days":"67", "times":
[{"from":"0:0","to":"17:59"}, {"from":"18:0","to":"23:59"}]}}
          </xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="version"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- propertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="propertyTypeType" use="required"/>
    <xs:attribute name="value" type="xs:string" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

```

```

<!-- propertyListType -->
<xs:complexType name="propertyListType">
  <xs:choice>
    <xs:sequence>
      <xs:element name="item" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="value" type="xs:string"
use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:element name="undefined" minOccurs="0">
      <xs:complexType/>
    </xs:element>
  </xs:choice>
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="type" type="propertyTypeType" use="required"/>
  <xs:attribute name="maxlen" type="xs:string" use="optional"/>
  <xs:attribute name="description" type="xs:string" use="optional"/>
</xs:complexType>
<!-- ssType -->
<xs:complexType name="ssType">
  <xs:sequence>
    <xs:element name="migrations" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="migration" type="migrationType"
minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="msd" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="property" type="propertyType"
minOccurs="0" maxOccurs="unbounded"/>
          <xs:element name="property_list"
type="propertyListType" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="ssd" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="property" type="propertyType"
minOccurs="0" maxOccurs="unbounded"/>
          <xs:element name="property_list"
type="propertyListType" minOccurs="0" maxOccurs="unbounded"/>

```

```

        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="csd" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="property" type="propertyType"
minOccurs="0" maxOccurs="unbounded"/>
          <xs:element name="property_list"
type="propertyListType" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="family" type="xs:string" use="optional">
    <xs:annotation>
      <xs:documentation xml:lang="en">
        Obsolete attribute. Use name instead of It.
      </xs:documentation>
      <xs:documentation xml:lang="ru">
        Устаревший атрибут. Теперь ДВО полностью определяется по
атрибуту name.
      </xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="version" type="xs:integer" use="required"/>
  <xs:attribute name="description" type="xs:string" use="optional"/>
  <xs:attribute name="enabled" type="xs:boolean" use="optional"/>
  <xs:attribute name="active" type="xs:boolean" use="optional"/>
</xs:complexType>
<!-- activatedSsType -->
<xs:complexType name="activatedSsType">
  <xs:complexContent>
    <xs:extension base="ssType">
      <xs:attribute name="domain" type="xs:string" use="required"/>
      <xs:attribute name="address" type="xs:string" use="required"/
>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<!-- rawSsType -->
<xs:complexType name="srcSsType">
  <xs:complexContent>
    <xs:extension base="ssType">
      <xs:attribute name="src" type="xs:string" use="required"/>
    </xs:extension>
  </xs:complexContent>

```



```

</xs:complexType>
<!-- ssListType -->
<xs:complexType name="ssListType">
  <xs:choice>
    <xs:sequence>
      <xs:element name="ss" type="ssType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:sequence>
      <xs:element name="activated_ss" type="activatedSsType"
minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:sequence>
      <xs:element name="src_ss" type="srcSsType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:choice>
</xs:complexType>
<!-- ss -->
<xs:element name="ss" type="ssType"/>
</xs:schema>

```

Нс_ss_acl_add

Команда позволяет добавить новые услуги в access-list определенного домена.

Аналог команды в CoCon:

```
cluster/storage/<CLUSTER>/ss/access-list add
```

Метод http запроса

POST

Шаблон http запроса

http://host:port/commands/ss_acl_add

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема xml файла с запросом

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="acl.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domains" type="domainsType"/>
            </xs:all>
            <xs:attribute name="storage" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- unknownSSNamesType -->
  <xs:complexType name="unknownSSNamesType">
    <xs:sequence>
      <xs:element name="ss" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="domains">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:choice>
                    <xs:element name="ok"/>
                    <xs:element name="unknown_ss_names"
type="unknownSSNamesType"/>
                  </xs:choice>
                  <xs:attribute name="name" type="xs:string"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:schema>

```

acl.xsd определен в разделе [Hc_ss - Управление услугами](#).

Пример:

Запрос: http://192.168.1.21:9999/commands/ss_acl_add

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="acl_add.xsd">
  <request storage="ds1">
    <domains>
      <domain name="biysk.local">
        <acl name="mcid"/>
        <acl name="acb"/>
      </domain>
      <domain name="ct.office">
        <acl name="acb"/>
        <acl name="rfc"/>
        <acl name="voice_page_control"/>
      </domain>
    </domains>
  </request>
</in>

```

Ответ:200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="acl_add.xsd">
  <domains>
    <domain name="biysk.local">
      <ok/>
    </domain>
    <domain name="biysk.test">
      <ok/>
    </domain>
  </domains>
</out>

```

Нс_ss_acl_remove

Команда позволяет удалить услуги из access-list определенного домена.

Аналог команды в CoCon:

```
/cluster/storage/<CLUSTER>/ss/access-list remove
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_acl_remove

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="acl.xsd"/>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domains" type="domainsType"/>
            </xs:all>
            <xs:attribute name="storage" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

acl.xsd определен в разделе [Нс_ss - Управление услугами](#).

Пример:

Запрос: http://192.168.1.21:9999/commands/ss_acl_remove

Параметры:

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="acl_remove.xsd" >
  <request storage="ds1">
    <domains>
      <domain name="biysk.local">
        <acl name="mcid"/>
        <acl name="acb"/>
      </domain>
      <domain name="ct.office">
        <acl name="acb"/>
        <acl name="rfc"/>
        <acl name="voice_page_control"/>
      </domain>
    </domains>
  </request>
</in>
```

Ответ:204

Нс_ss_acl_show

Команда позволяет вывести список услуг, доступных для определенного домена (виртуальной АТС).

Аналог команды в CoСon:

```
/cluster/storage/<CLUSTER>/ss/access-list show
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_acl_show

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:include schemaLocation="acl.xsd"/>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            <xs:attribute name="domain" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="domains" type="domainsType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

acl.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- aclType-->
  <xs:complexType name="aclType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- domainType -->
  <xs:complexType name="domainType">
    <xs:sequence>
      <xs:element name="acl" type="aclType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- domainsType -->
  <xs:complexType name="domainsType">
    <xs:sequence>
      <xs:element name="domain" type="domainType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>

```

Пример :

Запрос: http://192.168.1.21:9999/commands/ss_acl_show

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="acl_show.xsd">
  <request storage="ds1" domain="biysk.local"/>
</in>
```

Ответ:200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="acl_show.xsd">
  <domains>
    <domain name="biysk.local">
      <acl name="3way"/>
      <acl name="acb"/>
      <acl name="alarm"/>
      <acl name="assistant"/>
      <acl name="auto_redial"/>
      <acl name="auto_redial_with_callback"/>
      <acl name="avatar"/>
      <acl name="blf"/>
      <acl name="boss_group"/>
      <acl name="call_recording"/>
      <acl name="callback"/>
      <acl name="cc_agent"/>
      <acl name="cf_aon"/>
      <acl name="cf_sip"/>
      <acl name="cfb"/>
      <acl name="cfnr"/>
      <acl name="cfnr_type2"/>
      <acl name="cfos"/>
      <acl name="cft"/>
      <acl name="cfu"/>
      <acl name="cfu_type2"/>
      <acl name="cgg"/>
      <acl name="chold"/>
      <acl name="chunt"/>
      <acl name="click_to_call"/>
      <acl name="clip"/>
      <acl name="clir"/>
      <acl name="cliro"/>
      <acl name="cnip"/>
      <acl name="conference"/>
      <acl name="cpip"/>
      <acl name="ctr"/>
      <acl name="cw"/>
      <acl name="direct_call"/>
      <acl name="distinctive_picture"/>
      <acl name="distinctive_ring"/>
      <acl name="dnd"/>
      <acl name="fax_receiver"/>
      <acl name="fax_to_email"/>
      <acl name="fbc"/>
      <acl name="find_me"/>
      <acl name="find_me_no_response"/>
      <acl name="flexicall"/>
    </domain>
  </domains>
</out>
```



```
<acl name="follow_me"/>
<acl name="follow_me_no_response"/>
<acl name="intercom"/>
<acl name="intervention"/>
<acl name="mcid"/>
<acl name="meet_me_conference"/>
<acl name="message"/>
<acl name="ml"/>
<acl name="multicast_listen"/>
<acl name="multicast_page"/>
<acl name="my_number"/>
<acl name="park"/>
<acl name="personal_ivr"/>
<acl name="pickup"/>
<acl name="presence"/>
<acl name="qsig_cfb"/>
<acl name="qsig_cfnr"/>
<acl name="qsig_cfu"/>
<acl name="rbp"/>
<acl name="redial"/>
<acl name="remote_phone"/>
<acl name="rfc"/>
<acl name="ring_back_tone"/>
<acl name="sca"/>
<acl name="sco_black"/>
<acl name="sco_white"/>
<acl name="scr"/>
<acl name="speed_dial"/>
<acl name="teleconference"/>
<acl name="teleconference_manager"/>
<acl name="time_service"/>
<acl name="tsmn_request"/>
<acl name="vip_call"/>
<acl name="voice_page"/>
<acl name="voice_page_control"/>
<acl name="voicemail"/>
<acl name="walkie_talkie"/>
<acl name="zone_page"/>
</domain>
</domains>
</out>
```

Hc_ss_activate

Команда позволяет активировать услугу в определенном домене для определенного абонента (либо для всех абонентов).

Аналог команды в CoSop:

/cluster/storage/<CLUSTER>/ss/activate
/domain/<DOMAIN>/ss/activate

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_activate

XSD-схема XML-файла с запросом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="address" type="xs:string" use="optional"/>
    <xs:attribute name="ss" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="address" type="xs:string" use="optional"/>
    <xs:attribute name="ss" type="xs:string" use="optional"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- activateSsPropertyType -->
  <xs:complexType name="propertyType">
    <xs:attribute name="key" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- ssActivateType -->
  <xs:complexType name="ssActivateType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="ss" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="activate" type="ssActivateType"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>

```

```

        <xs:element name="ok" type="okType" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="error" type="errorType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
</xs:complexType>
</xs:element>
</xs:all>
</xs:complexType>
</xs:element>
</xs:schema>

```

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

Пример 1:

Запрос:

http://192.168.1.21:9999/commands/ss_activate

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_activate.xsd">
  <request domain="biysk.local">
    <activate ss="dnd" address="240465">
      <property value="[240301, 240464, 240001]" key="whitelist"/>
    </activate>
    <activate ss="cfu" address="240101">
      <property value="416977" key="number"/>
    </activate>
    <activate ss="cfnr" address="240470">
      <property value="89913699011" key="number"/>
      <property value="10" key="timeout"/>
    </activate>
  </request>
</in>

```

Ответ:

Код 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_activate.xsd">
  <response domain="biysk.local">
    <ok address="240470" ss="cfnr"/>
    <ok address="240101" ss="cfu"/>
    <ok address="240465" ss="dnd"/>
  </response>
</out>
```

Команда возвращает список услуг, которые администратор системы ECSS-10 сможет установить.

[Hc_ss_available_show](#)

Аналог команды в CoCon:

domain/<DOMAIN>/ss/available

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_avaliable_show

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="ss.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" minOccurs="1">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            </xs:complexType>
          </xs:element>
        </xs:all>
      </xs:complexType>
    </xs:element>
    <!-- out -->
    <xs:element name="out">
      <xs:complexType>
        <xs:all>
          <xs:element name="sss" type="ssListType" minOccurs="1" />
        </xs:all>
      </xs:complexType>
    </xs:element>
  </xs:schema>

```

ss.xsd:

Смотри параграф **hc_ss_installed**.

Пример::

Запрос: http://192.168.1.21:9999/commands/ss_avaliable_show

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_avaliable_show.xsd">
  <request storage="ds1"/>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_avaliable_show.xsd">
  <sss>
    <src_ss src="ss_zone_page.xml"
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="zone_page" version="10" description="Zone Page
      call" active="false" enabled="false">
      <msd>
        <property name="enumPriority" type="string" description="Call's priority"
          value="0_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
        <property name="enumInitiation_policy" type="string" description="Initiation
        call's policy" value="managers|members|any"/>
        <property name="integer_rangeMulticast_port" type="string"
        description="Internal field" value="[0,65535]"/>
        <property name="integer_rangeMulticast_ttl" type="string"
        description="Internal field" value="[1,255]"/>
        <property name="enumMulticast_codec" type="string" description="Internal
        field" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|G726-40"/>
      </msd>
      <ssd>
        <property name="dtmf_detector" type="boolean" description="" value="true"/>
      </ssd>
      <csd>
        <property name="priority" type="enum" description="(0_flash_override | 1_flash
        | 2_immediate | 3_priority | 4_routine)" value="3_priority"/>
        <property name="intro_pause" type="integer" description="Pause before play
        intro files" value="3"/>
        <property name="announcement_pause" type="integer" description="Pause before
        play announcement files" value="0"/>
        <property name="announcement_playback_cnt" type="integer"
        description="Announcement files playback count" value="1"/>
        <property name="announcement_repeat_time" type="integer"
        description="Announcement files repeat timeout" value="10"/>
        <property name="terminate_when_initiator_disconnects" type="boolean"
        description="Terminate call after initiator disconnected"/>
        <property name="terminate_when_playback_ends" type="boolean"
        description="Terminate call after playback ends"/>
        <property name="terminate_when_last_listener_leave" type="boolean"
        description="Terminate call after last listener disconnects"/>
        <property name="initiation_policy" type="enum" description="Initiation call's
        policy" value="any"/>
        <property name="display_name" type="string" description="ZonePage call
        DisplayName" value="ZonePage"/>
        <property name="play_release_tone" type="boolean" description="ZonePage
        release tone" value="false"/>
        <property name="announcement_repeat_digit" type="address" description="Digit
        to repeat announcement files" value="#" />

```

```

        <property name="multicast_ip" type="ip" description="Optional multicast
address to use for announcement" value="0.0.0.0"/>
        <property name="multicast_port" type="integer_range" description="Port of
multicast address" value="0"/>
        <property name="multicast_codec" type="enum" description="Codec to be used for
multicasting" value="PCMA"/>
        <property name="multicast_ttl" type="integer_range" description="TTL for
multicast" value="1"/>
        <property_list name="members" type="address" maxlen="256" description="Zone
page member list">
            <undefined/>
        </property_list>
        <property_list name="managers" type="address" maxlen="256" description="Zone
page managers list"/>
        <property_list name="intro_files" type="media_resource" maxlen="10"
description="List of Intro media_files"/>
        <property_list name="announcement_files" type="media_resource" maxlen="10"
description="List of Announcement media_files"/>
    </csd>
</src_ss>
<src_ss src="ss_walkie_talkie.xml"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="walkie_talkie" version="1" description="Walkie
Talkie" active="false" enabled="false">
    <ssd>
        <property name="dtmf_detector" type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="swap_digit" type="address" description="Digit to swap
reception / transmission mode" value="*"/>
    </csd>
</src_ss>
<src_ss src="ss_voicemail.xml"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="voicemail" version="13" description="Voice mail
service" active="false" enabled="false">
    <msd>
        <property_list name="conflict_list" type="atom" maxlen="0" description="List
of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
            <item value="cfu"/>
            <item value="find_me"/>
            <item value="cf_aon"/>
            <item value="cft"/>
        </property_list>
    </msd>
</ssd>

```



```

        <property name="dtmf_detector" type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="no_reply" type="boolean" description="Activate service when
subscriber no reply" value="true"/>
        <property name="busy" type="boolean" description="Activate service when
subscriber is busy" value="false"/>
        <property name="dnd" type="boolean" description="Activate service when
subscriber activate DND" value="false"/>
        <property name="out_of_service" type="boolean" description="Activate service
when subscriber is out of service" value="true"/>
        <property name="unconditional" type="boolean" description="Activate service
any time" value="false"/>
        <property name="timeout" type="positive_integer" description="Timeout before
call will be forwarded at voicemail (sec)" value="30"/>
        <property name="send_by_email" type="boolean" description="Send recorded
message by email" value="false"/>
        <property name="min_message_duration" type="positive_integer" description="The
minimum voice mail duration (sec). In case of voice mail duration will be less that
min_message_duration, this message will be dropped" value="0"/>
        <property name="max_message_duration" type="positive_integer" description="The
maximum voice mail duration (sec). 0 - in case of not limited duration" value="300"/>
        <property name="mailbox_volume" type="positive_integer" description="The
mailbox volume (sec)" value="1800"/>
        <property name="override_messages" type="boolean" description="Override
messages in case of mailbox is full" value="true"/>
        <property name="email_from" type="string" description="Email from template.
Macro variables: %DATE%, %TIME%, %CALLING%, %DISPLAYNAME%" value="%DISPLAYNAME%"/>
        <property name="email_subject" type="string" description="Email subject
template. Macro variables: %DATE%, %TIME%, %CALLING%, %DISPLAYNAME%" value="Voicemail from
%CALLING% at %DATE% %TIME%"/>
        <property name="email_body" type="string" description="Email body template.
Macro variables: %DATE%, %TIME%, %CALLING%, %DISPLAYNAME%" value=""/>
        <property name="play_message_details" type="boolean" description="Play
message's details (number, time) before message" value="false"/>
        <property name="password" type="address" description="There is password to
have access to the voicemail box" value=""/>
        <property name="max_silence" type="positive_integer" description="There is
time(sec) while service wait for a contiguous period of silence before terminating an incoming
call to voice mail" value="0"/>
    </csd>
</src_ss>
<src_ss src="ss_voice_page_control.xml"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="voice_page_control" version="2"
description="'Voice Page Control' service" active="false" enabled="false">
    <csd>
        <property name="accept_incoming_vp" type="boolean" description="Accept or
Reject voice page calls" value="true"/>
    </csd>
</src_ss>

```

```

        <property name="block_incoming_vp_by_dnd" type="boolean" description="Block
voice page calls by DND service" value="true"/>
        <property name="mute_incoming_vp" type="boolean" description="Accept incoming
voice page call in silent mode" value="true"/>
    </csd>
</src_ss>
<src_ss src="ss_voice_page.xml"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="voice_page" version="9" description="Voice Page
call" active="false" enabled="false">
    <msd>
        <property name="enumPriority" type="string" description="Call's priority"
value="0_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
    </msd>
    <ssd>
        <property name="dtmf_detector" type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="skip_cf" type="boolean" description="Skipping call forwarding"
value="true"/>
        <property name="priority" type="enum" description="(0_flash_override | 1_flash
| 2_immediate | 3_priority | 4_routine)" value="3_priority"/>
        <property name="announcement_repeat_time" type="integer"
description="Announcement files repeat timeout" value="10"/>
        <property name="announcement_playback_cnt" type="integer"
description="Announcement files playback count [0..5]" value="0"/>
        <property name="swap_digit" type="address" description="Digit to swap
reception / transmission mode" value="*/>
        <property name="announcement_repeat_digit" type="address" description="Digit
to repeat announcement files" value="#"/>
        <property_list name="intro_files" type="media_resource" maxlen="10"
description="List of Intro media_files"/>
        <property_list name="announcement_files" type="media_resource" maxlen="10"
description="List of Announcement media_files"/>
    </csd>
</src_ss>
<src_ss src="ss_vip_call.xml"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="vip_call" version="4" description="Vip call"
active="false" enabled="false">
    <msd>
        <property name="enumPriority" type="string" description="Call's priority"
value="0_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
    </msd>
    <csd>
        <property name="priority" type="enum" description="(0_flash_override | 1_flash
| 2_immediate | 3_priority | 4_routine)" value="4_routine"/>
        <property name="implicit_mode" type="boolean" description="Use vip call
without feature code." value="false"/>

```

```

        <property name="skip_low_priority_call" type="boolean" description="Skipping
second call if he has with low priority." value="false"/>
    </csd>
</src_ss>
<src_ss src="ss_tsmn_request.xml"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="tsmn_request" version="3" description="Support
request on TSMN" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available values:
user | trunk | any" value="trunk"/>
    </msd>
    <csd>
        <property name="timeout" type="positive_integer" description="Timeout on TSMN
request(s) (msec)" value="3000"/>
        <property name="tsmn_trunk" type="trunk" description="Trunk to TSMN server"/>
        <property name="tsmn_trunk_backup" type="trunk" description="Backup trunk to
TSMN server" value=""/>
    </csd>
</src_ss>
<src_ss src="ss_teleconference_manager.xml"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="teleconference_manager" version="3"
description="Teleconference manager" active="false" enabled="false">
    <csd>
        <property_list name="second_line" type="address" maxlen="10"
description="Second line numbers"/>
    </csd>
</src_ss>
<src_ss src="ss_teleconference.xml"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="teleconference" version="4"
description="Teleconference" active="false" enabled="false">
    <msd>
        <property name="enumType" type="string" description="Teleconference type:
meeting or room" value="meeting|room"/>
    </msd>
    <csd>
        <property name="destroy_by_no_initiator" type="boolean" description="Destroy
the teleconference by release from the initiator?" value="true"/>
        <property name="template_from_room" type="teleconference_room"
description="Get template from the teleconference room" value=""/>
        <property name="type" type="enum" description="Teleconference type: meeting or
room" value="meeting"/>
    </csd>
</src_ss>
<src_ss src="ss_speed_dial.xml"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="speed_dial" version="5" description="Speed dial
(Abbreviated dialing)" active="false" enabled="false">

```

```

    <msd>
      <property name="enumType" type="string" description="The type of address book.
address_book_10 supports 10 number in the address book. address_book_100 supports 100 numbers
on the address book" value="address_book_10|address_book_100"/>
    </msd>
  </csd>
  <src_ss>
    <src_ss src="ss_smart_cancel.xml"
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="smart_cancel" version="2" description="Smart
cancel service" active="false" enabled="false">
      <msd>
        <property name="integer_rangeTimeout" type="string" description="Timeout range
property" value="[0,5]"/>
      </msd>
      <csd>
        <property name="timeout" type="integer_range" description="Timeout to wait
release" value="2"/>
      </csd>
    </src_ss>
    <src_ss src="ss_sip_message.xml"
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="sip_message" version="1" description="SIP Message
service" active="false" enabled="false">
      <msd>
        <property name="enumMode" type="string" description="Call's priority"
value="sendonly|recvonly|sendrecv"/>
      </msd>
      <csd>
        <property name="mode" type="enum" description="(sendonly | recvonly |
sendrecv)" value="sendrecv"/>
      </csd>
    </src_ss>
    <src_ss src="ss_second_handset.xml"
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="second_handset" version="1" description="Second
Handset" active="false" enabled="false"/>
      <src_ss src="ss_scr.xml"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="scr" version="2" description="Selective Call
Rejection" active="false" enabled="false">
        <csd>
          <property_list name="blacklist" type="index_address" maxlen="50"
description="Black list members. Format: <address>[/<index>]"/>
          <undefined/>
        </csd>
      </src_ss>
    </src_ss>
  </src_ss>

```

```

        </property_list>
    </csd>
</src_ss>
<src_ss src="ss_sco_white.xml"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="sco_white" version="2" description="Selective
Call Originating, White list" active="false" enabled="false">
    <csd>
        <property_list name="whitelist" type="index_address" maxlen="50"
description="White list members. Format: <address>[/<index>]">
            <undefined/>
        </property_list>
    </csd>
</src_ss>
<src_ss src="ss_sco_black.xml"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="sco_black" version="2" description="Selective
Call Originating, Black list" active="false" enabled="false">
    <csd>
        <property_list name="blacklist" type="index_address" maxlen="50"
description="Black list members. Format: <address>[/<index>]">
            <undefined/>
        </property_list>
    </csd>
</src_ss>
<src_ss src="ss_sca.xml"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="sca" version="2" description="Selective Call
Acceptance" active="false" enabled="false">
    <csd>
        <property_list name="whitelist" type="index_address" maxlen="50"
description="White list members. Format: <address>[/<index>]">
            <undefined/>
        </property_list>
    </csd>
</src_ss>
<src_ss src="ss_ring_back_tone.xml"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="ring_back_tone" version="2" description="Ring
back tone" active="false" enabled="false">
    <csd>
        <property name="file" type="media_resource" description="File to play on
ring"/>
    </csd>
</src_ss>
<src_ss src="ss_rfc.xml"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="rfc" version="2" description="Rejection of
Forwarded Calls" active="false" enabled="false"/>
</src_ss src="ss_remote_ring.xml"

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="remote_ring" version="1" description="Remote ring
to callee" active="false" enabled="false">
        <msd>
            <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
                <item value="distinctive_ring"/>
            </property_list>
        </msd>
        <csd>
            <property_list name="announcement_files" type="media_resource"
maxlen="10" description="List of Announcement media_files"/>
        </csd>
    </src_ss>
    <src_ss src="ss_remote_phone.xml"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="remote_phone" version="1" description="Remote
phone" active="false" enabled="false">
        <msd>
            <property name="handle_module" type="atom" description=""
value="ss_remote_phone_handler"/>
        </msd>
        <csd>
            <property name="remote_endpoint" type="interface" description="Remote
interface" value=""/>
            <property name="pin" type="address" description="PIN-code"/>
        </csd>
    </src_ss>
    <src_ss src="ss_redial.xml"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="redial" version="2" description="Last number
redial" active="false" enabled="false"/>
        <src_ss src="ss_rbp.xml"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="rbp" version="3" description="SS for restricting
call-processes by NI and PIN" active="false" enabled="false">
            <csd>
                <property name="implicit_call" type="boolean" description="Use
implicit RBP call (Request PIN-code in case of use recsricted call)" value="false"/>
                <property_list name="restricted_ni" type="ni" maxlen="6"
description="Restricted number indication">
                    <undefined/>
                </property_list>
            </csd>
        </src_ss>
        <src_ss src="ss_qsig_cfu.xml"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfu" version="3" description="QSIG: Call
Forwarding Unconditional" active="false" enabled="false">
            <msd>

```

```

        <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="trunk"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="follow_me"/>
        </property_list>
    </msd>
    <csd>
        <property name="hide_forwarding" type="boolean" description="Hide
forwarding fields (rgn, ocdrn)" value="false"/>
    </csd>
</src_ss>
<src_ss src="ss_qsig_cfnr.xml"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfnr" version="2" description="QSIG: Call
Forwarding No Reply" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available
values:user | trunk | any" value="trunk"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="cfu"/>
        </property_list>
    </msd>
</src_ss>
<src_ss src="ss_qsig_cfb.xml"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfb" version="3" description="QSIG: Call
Forwarding Busy" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="trunk"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="cfu"/>
        </property_list>
    </msd>
    <csd>
        <property name="hide_forwarding" type="boolean" description="Hide
forwarding fields (rgn, ocdrn)" value="false"/>
    </csd>
</src_ss>
<src_ss src="ss_privacy.xml"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="privacy" version="2" description="Privacy"
active="false" enabled="false">
    <msd>
        <property name="enumMode" type="string" description="Mode"
value="on_demand|always_on"/>

```

```

        </msd>
        <ssid>
            <property name="dtmf_detector" type="boolean" description=""
value="true"/>
        </ssid>
        <csd>
            <property name="mode" type="enum" description="Privacy usage mode"
value="on_demand"/>
        </csd>
    </src_ss>
    <src_ss src="ss_presence.xml"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="presence" version="2" description="SIP Presence"
active="false" enabled="false">
        <csd>
            <property name="max_subscribe" type="positive_integer"
description="Maximum number of subscriptions" value="16"/>
        </csd>
    </src_ss>
    <src_ss src="ss_pickup.xml"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="pickup" version="2" description="Call Pickup"
active="false" enabled="false">
        <csd>
            <property_list name="pickup_groups" type="index_string"
maxlen="10" description="Call pick-up's groups names. Format: <Name>[/<Inx>]"/>
            <undefined/>
        </property_list>
        </csd>
    </src_ss>
    <src_ss src="ss_personal_ivr.xml"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="personal_ivr" version="2" description="Personal
IVR script" active="false" enabled="false">
        <msd>
            <property name="handle_module" type="atom" description=""
value="ss_personal_ivr_handler"/>
            <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="user"/>
        </msd>
        <csd>
            <property name="script" type="ivr_script_id" description="Script
ID" value=""/>
        </csd>
    </src_ss>
    <src_ss src="ss_park.xml"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="park" version="2" description="Call park"
active="false" enabled="false"/>
    <src_ss src="ss_multiline.xml"

```



```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="m1" version="4" description="Multiline"
        active="false" enabled="false">
            <msd>
                <property name="integer_rangeLine_count" type="string"
                description="Internal field" value="[0,16]"/>
                <property name="integer_rangeLine_count_out" type="string"
                description="Internal field" value="[-1,16]"/>
            </msd>
            <csd>
                <property name="line_count" type="integer_range"
                description="Line count" value="3"/>
                <property name="line_count_out" type="integer_range"
                description="Outgoing lines count" value="-1"/>
            </csd>
        </src_ss>
        <src_ss src="ss_multicast_listen.xml"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="multicast_listen" version="1"
        description="Multicast Listen Service" active="false" enabled="false">
            <msd>
                <property name="enumCodec" type="string" description="Internal
                field to codec property" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|
                G726-40"/>
                <property name="integer_rangePort" type="string"
                description="Internal field to udp port property" value="[10000,65535]"/>
            </msd>
            <csd>
                <property name="ip" type="ip" description="Ip address for
                listen multicast."/>
                <property name="port" type="integer_range" description="Port
                for listen multicast (10000-65535)."/>
                <property name="codec" type="enum" description="Audio codec
                for multicast traffic." value="G722"/>
            </csd>
        </src_ss>
        <src_ss src="ss_message.xml"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="message" version="2" description="Message
        service" active="false" enabled="false">
            <msd>
                <property name="owner" type="atom" description="SS type.
                Available values: user | trunk | any" value="any"/>
            </msd>
            <ssd>
                <property name="message_box_size" type="integer"
                description="Message box size" value="20"/>
                <property name="remove_only_after_answer" type="boolean"
                description="Remove message only after abonent answer" value="false"/>
            </ssd>

```

```

        </src_ss>
        <src_ss src="ss_meet_me_conference.xml"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="meet_me_conference" version="2" description="Meet
            Me Conference" active="false" enabled="false">
            <msd>
                <property name="regimeMode" type="string"
                    description="Internal field to validate mode property" value="master:destroy_mode:room_number|
                    member:room_number"/>
                <property name="enumDestroy_mode" type="string"
                    description="Internal field to validate mode property" value="by_no_master|by_no_more_calls"/>
                <property_list name="conflict_list" type="atom" maxlen="0"
                    description="List of SS that is conflict with current SS">
                    <item value="assistant"/>
                    <item value="follow_me"/>
                    <item value="cfu_type2"/>
                </property_list>
            </msd>
            <csd>
                <property name="mode" type="regime" description="The mode of
                    meet me conference (master|member)"/>
                <property name="destroy_mode" type="enum" description="The
                    mode of destroying conference (by_no_master|by_no_more_calls)" value="by_no_more_calls"/>
                <property name="room_number" type="address" description="Room
                    number" value=""/>
            </csd>
        </src_ss>
        <src_ss src="ss_mcast_pg.xml"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="multicast_page" version="1"
            description="Multicast IP Paging" active="false" enabled="false">
            <msd>
                <property name="enumCodec" type="string" description="Internal
                    field" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|G726-40"/>
                <property name="integer_rangePort" type="string"
                    description="Internal field" value="[10000,65535]"/>
                <property name="integer_rangeTtl" type="string"
                    description="Internal field" value="[1,255]"/>
            </msd>
            <csd>
                <property name="ip" type="ip" description="IP address to cast
                    the media to"/>
                <property name="port" type="integer_range" description="UDP
                    port to cast the media to (10000-65535)"/>
                <property name="codec" type="enum" description="Codec to be
                    used for multicasting" value="PCMA"/>
                <property name="ttl" type="integer_range" description="TTL for
                    multicast packets (1-255)" value="1"/>
            </csd>
        </src_ss>

```

```

        <src_ss src="ss_malicious_call.xml"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="mcid" version="2" description="Malicious Call
            Identification" active="false" enabled="false"/>
        <src_ss src="ss_introduce.xml"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="introduce" version="1" description="Introduce
            announcement from callee" active="false" enabled="false">
            <csd>
                <property name="announcement" type="media_resource"
                description="Announcement media file"/>
            </csd>
        </src_ss>
        <src_ss src="ss_intervention.xml"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="intervention" version="4"
            description="Intervention" active="false" enabled="false">
            <msd>
                <property name="enumRestriction" type="string"
                description="Internal field to validate mode property" value="full_control|
            implicitly_allowed|explicitly_allowed"/>
            </msd>
            <ssd>
                <property name="dtmf_detector" type="boolean"
            description="" value="true"/>
            </ssd>
            <csd>
                <property name="restriction" type="enum" description="The
            restriction to the intervention call (full_control|allowed_only)" value="full_control"/>
            </csd>
        </src_ss>
        <src_ss src="ss_intercom.xml"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="intercom" version="1" description="Intercom"
            active="false" enabled="false"/>
        <src_ss src="ss_hide_cf_name.xml"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="hide_cf_name" version="1" description="Hide call
            forwarding name for subscriber A" active="false" enabled="false"/>
        <src_ss src="ss_fxo_trunk.xml"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="fxo_trunk" version="1" description="FX0 trunk
            service" active="false" enabled="false">
            <msd>
                <property name="owner" type="atom" description="SS
            type. Available values: user | trunk | any" value="trunk"/>
            </msd>
        </src_ss>
        <src_ss src="ss_follow_me_no_response.xml"

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="follow_me_no_response" version="8"
description="Follow me no response" active="false" enabled="false">
    <msd>
        <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="cft"/>
            <item value="cgg"/>
            <item value="follow_me"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address"
description="Followed number" value=""/>
        <property name="timeout" type="positive_integer"
description="Timeout before call forward (sec)" value="20"/>
        <property name="pin" type="address"
description="PIN-code" value=""/>
        <property name="external_number" type="address"
description="External forward number" value=""/>
        <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
    </csd>
</src_ss>
<src_ss src="ss_follow_me.xml"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="follow_me" version="11" description="Follow me"
        active="false" enabled="false">
    <msd>
        <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
            <item value="assistant"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address"
description="Followed number" value=""/>
        <property name="pin" type="address"
description="PIN-code" value=""/>

```

```

        <property name="external_number" type="address"
description="External forward number" value=""/>
        <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
    </csd>
</src_ss>
<src_ss src="ss_flexicall.xml"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="flexicall" version="1" description="FlexiCall"
active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS
type. Available values: user | trunk | any." value="user"/>
        <property name="enumMode" type="string"
description="Internal field to validate mode property" value="all|external|internal"/>
        <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
            <item value="cgg"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="mode" type="enum"
description="Mode operating. Available values (all | external | internal)"/>
        <property_list name="external_numbers"
type="address" maxlen="10" description="Group's external numbers"/>
        <property_list name="internal_numbers"
type="address" maxlen="10" description="Group's internal numbers"/>
    </csd>
</src_ss>
<src_ss src="ss_find_me_no_response.xml"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="find_me_no_response" version="6"
description="Find Me no response" active="false" enabled="false">
    <msd>
        <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="ctu"/>
            <item value="cgg"/>
            <item value="follow_me"/>
            <item value="follow_me_no_response"/>
            <item value="find_me"/>
        </property_list>
    </msd>
</ssd>

```

```

        <property name="forwarding_ss" type="boolean"
description="" value="false"/>
    </ssd>
    <csd>
        <property name="send_call_diversion"
type="boolean" description="Send call diversion flag in call signalisation" value="true"/>
        <property name="timeout" type="positive_integer"
description="Timeout before call forward (sec)" value="20"/>
        <property_list name="groups" type="find_me_group"
maxlen="32" description="Find me groups"/>
    </csd>
</src_ss>
<src_ss src="ss_find_me.xml"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="find_me" version="7" description="Find Me"
active="false" enabled="false">
    <msd>
        <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
            <item value="cfu"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="send_call_diversion"
type="boolean" description="Send call diversion flag in call signalisation" value="true"/>
        <property_list name="groups" type="find_me_group"
maxlen="32" description="Find me groups"/>
    </csd>
</src_ss>
<src_ss src="ss_fbc.xml"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="fbc" version="1" description="Forwarding Barring
Call" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS
type. Available values: user | trunk | any" value="any"/>
    </msd>
</src_ss>
<src_ss src="ss_fax_to_email.xml"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="fax_to_email" version="2" description="Fax to
email" active="false" enabled="false">

```

```

        <csd>
            <property name="email" type="string"
description="Email address" value=""/>
            <property name="send_error_report" type="boolean"
description="Send email in case of fax received failed" value="false"/>
        </csd>
    </src_ss>
    <src_ss src="ss_fax_receiver.xml"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="fax_receiver" version="2" description="Enable fax
receiving ability for terminals which is not support fax by their own" active="false"
enabled="false"/>
        <src_ss src="ss_dnd.xml"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="dnd" version="4" description="Do not
Disturb" active="false" enabled="false">
            <msd>
                <property name="enumMode" type="string"
description="Internal field to validate mode property" value="default|smart"/>
            </msd>
            <csd>
                <property name="mode" type="enum"
description="The mode of the working dnd service (default|smart)" value="default"/>
                <property_list name="whitelist"
type="index_address" maxlen="10" description="White list members. Format: <address>[</index>]"
/>
            </csd>
        </src_ss>
        <src_ss src="ss_distinctive_ring.xml"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="distinctive_ring" version="4"
description="Destination ringtone change" active="false" enabled="false">
            <csd>
                <property_list name="filters"
type="distinctive_ring" maxlen="16" description="Distinctive ring for specific numbers"/>
            </csd>
        </src_ss>
        <src_ss src="ss_distinctive_picture.xml"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="distinctive_picture" version="2"
description="Display caller image by groups" active="false" enabled="false">
            <csd>
                <property_list name="groups"
type="distinctive_picture" maxlen="20" description="Distinctive picture for specific numbers"/
>
            </csd>
        </src_ss>
    </src_ss src="ss_direct_call.xml"

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="direct_call" version="2"
description="Direct call" active="false" enabled="false">
    <csd>
        <property name="number" type="address"
description="Call number to direct call"/>
        <property name="timeout"
type="positive_integer" description="Timeout before direct call (sec)" value="5"/>
    </csd>
</src_ss>
<src_ss src="ss_cw.xml"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cw" version="8" description="Call
Waiting" active="false" enabled="false">
    <msd>
        <property name="enumPlay_voice_message"
type="string" description="Internal field to validate play_voice_message property"
value="true|false|default"/>
        <property_list name="dependency_list"
type="atom" maxlen="0" description="List of SS that is depended from SS">
            <item value="chold"/>
        </property_list>
    </msd>
    <ssd>
        <property name="dtmf_detector" type="boolean"
description="" value="true"/>
    </ssd>
</csd>
</src_ss>
<src_ss src="ss_ctr.xml"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="ctr" version="5" description="Call
Transfer" active="false" enabled="false">
    <msd>
        <property_list name="dependency_list"
type="atom" maxlen="0" description="List of SS that is depended from SS">
            <item value="chold"/>
        </property_list>
    </msd>
    <ssd>
        <property name="dtmf_detector" type="boolean"
description="" value="true"/>
    </ssd>
</src_ss>
<src_ss src="ss_cpip.xml"

```



```

xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cpip" version="2" description="Calling
Picture Identification Presentation" active="false" enabled="false">
  <msd>
    <property name="enumDisplay_mode"
type="string" description="Internal field to validate 'display-mode' property" value="icon|
fullscreen"/>
  </msd>
  <csd>
    <property name="display_mode" type="enum"
description="Sets the picture display mode: icon mode or fullscreen mode." value="icon"/>
  </csd>
</src_ss>
<src_ss src="ss_conference.xml"
xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="conference" version="9"
description="Conference Call, Add-on" active="false" enabled="false">
  <msd>
    <property name="enumDestroy_mode"
type="string" description="Internal field to validate mode property" value="by_no_master|
by_no_more_calls"/>
    <property_list name="dependency_list"
type="atom" maxlen="0" description="List of SS that is depended from SS">
      <item value="chold"/>
      <item value="ctr"/>
    </property_list>
    <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
      <item value="3way"/>
    </property_list>
  </msd>
  <csd>
    <property name="destroy_mode" type="enum"
description="The mode of destroying conference (by_no_master|by_no_more_calls)"
value="by_no_master"/>
    <property name="max_participants"
type="positive_integer" description="The maximum number of participants in the conference,
limited to the number 64 at the top. By default 16" value="16"/>
  </csd>
</src_ss>
<src_ss src="ss_cnip.xml"
xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cnip" version="3" description="Calling
Name Identification Presentation" active="false" enabled="false">
  <msd>
    <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
  </msd>
</src_ss>
<src_ss src="ss_cliro.xml"

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cliro" version="2" description="Calling
Line Identification Restriction Override" active="false" enabled="false"/>
        <src_ss src="ss_clir.xml"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="clir" version="3" description="Calling
Line Identification Restriction" active="false" enabled="false">
        <msd>
        <property name="enumMode" type="string"
description="Internal field to validate mode property" value="user_defined|restrict_all"/>
        </msd>
        <csd>
        <property name="mode" type="enum"
description="The mode of the working clir service (user_defined|restrict_all)"
value="restrict_all"/>
        </csd>
        </src_ss>
        <src_ss src="ss_clip.xml"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="clip" version="5" description="Calling
Line Identification Presentation" active="false" enabled="false"/>
        <src_ss src="ss_click_to_call.xml"
        xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="click_to_call" version="5"
description="Click to call" active="false" enabled="false">
        <csd>
        <property name="number" type="address"
description="User's number"/>
        <property name="auto_answer"
type="boolean" description="Auto answer" value="true"/>
        <property_list name="hashes"
type="string" maxlen="10" description="User's hashes">
        <undefined/>
        </property_list>
        </csd>
        </src_ss>
        <src_ss src="ss_chunt.xml"
        xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="chunt" version="7"
description="Call hunt groups" active="false" enabled="false">
        <msd>
        <property name="regimeMode"
type="string" description="Internal field to validate search_strategy property"
value="group:numbers:queue_strategy|
serial:numbers:cyclic:search_strategy>window_shift_timeout:max_shift_count:queue_strategy|
longest_idle:numbers:cyclic>window_shift_timeout:max_shift_count:queue_strategy|
delay_group:numbers>window_shift_timeout:queue_strategy|manual"/>
        <property name="enumSearch_strategy"
type="string" description="Internal field to validate search_strategy property" value="first|
last|random|longest_idle"/>

```

```

        <property name="enumQueue_strategy"
type="string" description="Internal field to validate queue_strategy property" value="wait|
drop"/>
        <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
            <item value="cfu"/>
            <item value="find_me"/>
            <item value="cf_aon"/>
            <item value="cft"/>
            <item value="cgg"/>
        </property_list>
    </msd>
    <ssd>
        <property name="delay_retrieve_timeout
_1" type="integer" description="Delay retrieve timeout for release group call by group
subscriber" value="1000"/>
        <property name="delay_retrieve_timeout
_2" type="integer" description="Delay retrieve timeout for release group call by slave
subscriber" value="5000"/>
        <property name="delay_retrieve_timeout
_3" type="integer" description="Delay retrieve timeout for waiting free group subscriber"
value="15000"/>
        <property name="forwarding_ss"
type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="mode" type="regime"
description="The type of the chunt"/>
        <property name="cyclic" type="boolean"
description="Flag that is show is call hunt group cyclic or not" value="false"/>
        <property name="reset_window_start_pos
_after_call" type="boolean" description="Reset window start position after all incoming calls
will be connected with group calls" value="false"/>
        <property name="window_shift_timeout"
type="positive_integer" description="Interval after which window shift on the window_shift
value" value="0"/>
        <property name="max_shift_count"
type="integer" description="The number of times that a window can be shift" value="-1"/>
        <property name="window_size"
type="positive_integer" description="The number of simultaneous calls to the call-hunt's
subscribers" value="1"/>
        <property name="window_start_pos"
type="integer" description="The index (zero-based) of the first subscriber in the window"
value="0"/>

```

```

                                <property name="window_shift"
type="positive_integer" description="The value on which shift window_start_pos when
window_shift_timeout fired" value="1"/>
                                <property name="queue_strategy"
type="enum" description="(wait | drop)" value="wait"/>
                                <property name="search_strategy"
type="enum" description="(first | last | random | longest_idle)" value="first"/>
                                <property name="queue_size"
type="positive_integer" description="Wait calls' queue size" value="0"/>
                                <property_list name="numbers"
type="address" maxlen="10" description="Call-hunt member list">
                                    <undefined/>
                                </property_list>
                                </csd>
                            </src_ss>
                            <src_ss src="ss_chold.xml"
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="chold" version="8"
                                description="Call hold" active="false" enabled="false">
                                    <ssd>
                                        <property name="dtmf_detector"
type="boolean" description="" value="true"/>
                                        </ssd>
                                    </csd>
                                        <property name="dtmf_sequence_as_flash"
" type="boolean" description="Try interpret the DTMF sequence as a flash (look at incall
feature codes)" value="false"/>
                                        <property name="disable_moh"
type="boolean" description="Disable music on hold" value="false"/>
                                        </csd>
                                    </src_ss>
                                    <src_ss src="ss_cgg.xml"
                                        xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cgg" version="7"
                                        description="Group-calling" active="false" enabled="false">
                                            <msd>
                                                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                                    <item value="assistant"/>
                                                    <item value="follow_me"/>
                                                    <item value="cfu_type2"/>
                                                    <item value="meet_me_conference"/>
                                                    <item value="cfu"/>
                                                    <item value="find_me"/>
                                                    <item value="cf_aon"/>
                                                    <item value="cft"/>
                                                </property_list>
                                            </msd>
                                        </ssd>

```

```

        <property name="forwarding_ss"
type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="send_call_diversion"
type="boolean" description="Send call diversion flag in call signalisation" value="true"/>
        <property_list name="numbers"
type="address" maxlen="10" description="Group's numbers">
            <undefined/>
        </property_list>
    </csd>
</src_ss>
<src_ss src="ss_cfu_type2.xml"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfu_type2" version="14"
description="Call Forwarding Unconditional without notification" active="false"
enabled="false">
    <msd>
        <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
        <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss"
type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address"
description="Forward number"/>
        <property name="use_redirection_counte
r" type="boolean" description="Use redirectionCounter field to count hops" value="false"/>
        <property name="external_number"
type="address" description="External forward number" value=""/>
        <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
    </csd>
</src_ss>
<src_ss src="ss_cfu.xml"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfu" version="14"
description="Call Forwarding Unconditional" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
        <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">

```

```

        <item value="assistant"/>
        <item value="follow_me"/>
        <item value="cfu_type2"/>
        <item value="meet_me_conference"/>
    </property_list>
</msd>
<ssd>
    <property name="forwarding_ss"
type="boolean" description="" value="true"/>
    </ssd>
<csd>
    <property name="number" type="address"
description="Forward number"/>
    <property name="external_number"
type="address" description="External forward number" value=""/>
    <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
    </csd>
</src_ss>
<src_ss src="ss_cft.xml"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cft" version="14"
description="Call Forwarding Unconditional Time Dependent" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
        <property name="enumTime_format"
type="string" description="Internal field to validate time_format property" value="local|utc"/
>
        <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
            <item value="cfu"/>
            <item value="find_me"/>
            <item value="cf_aon"/>
        </property_list>
    </msd>
<ssd>
    <property name="forwarding_ss"
type="boolean" description="" value="true"/>
    </ssd>
<csd>
    <property name="day_1" type="term"
description="{[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>

```

```

                                <property name="day_2" type="term"
description="{[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                <property name="day_3" type="term"
description="{[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                <property name="day_4" type="term"
description="{[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                <property name="day_5" type="term"
description="{[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                <property name="day_6" type="term"
description="{[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                <property name="day_7" type="term"
description="{[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                <property name="time_format"
type="enum" description="CFT for Sunday (local | utc)" value="local"/>
                                </csd>
                                </src_ss>
                                <src_ss src="ss_cfos.xml"
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfos" version="12"
description="Call Forwarding Out of Service" active="false" enabled="false">
                                <msd>
                                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
                                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                <item value="cfu"/>
                                <item value="assistant"/>
                                <item value="follow_me"/>
                                <item value="tsmn_request"/>
                                </property_list>
                                </msd>
                                <ssd>
                                <property name="forwarding_ss"
type="boolean" description="" value="true"/>
                                </ssd>
                                </csd>
                                <property name="number" type="address"
description="Forward number"/>
                                <property name="external_number"
type="address" description="External forward number" value=""/>
                                <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
                                </csd>
                                </src_ss>

```

```

        <src_ss src="ss_cfnr_type2.xml"
            xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfnr_type2" version="11"
            description="Call Forwarding No Reply without notification" active="false" enabled="false">
            <msd>
                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                    <item value="cfu"/>
                    <item value="assistant"/>
                    <item value="cfnr"/>
                    <item value="follow_me"/>
                    <item value="tsmn_request"/>
                </property_list>
            </msd>
            <ssd>
                <property name="forwarding_ss"
type="boolean" description="" value="true"/>
            </ssd>
            <csd>
                <property name="number" type="address"
description="Forward number"/>
                <property name="timeout"
type="positive_integer" description="Timeout before call forward (sec)" value="15"/>
                <property name="use_redirection_counter" type="boolean" description="Use redirectionCounter field to count hops" value="false"/>
                <property name="external_number"
type="address" description="External forward number" value=""/>
                <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
            </csd>
        </src_ss>
        <src_ss src="ss_cfnr.xml"
            xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfnr" version="11"
            description="Call Forwarding No Reply" active="false" enabled="false">
            <msd>
                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                    <item value="cfu"/>
                    <item value="assistant"/>
                    <item value="follow_me"/>
                    <item value="tsmn_request"/>
                </property_list>
            </msd>
            <ssd>

```



```

type="boolean" description="" value="true"/>
                                <property name="forwarding_ss"
                                </ssd>
                                <csd>
                                <property name="number" type="address"
                                <property name="external_number"
type="address" description="External forward number" value=""/>
                                <property name="use_external_number"
                                <property name="timeout"
type="positive_integer" description="Timeout before call forward (sec)" value="15"/>
                                </csd>
                                </src_ss>
                                <src_ss src="ss_cfb.xml"
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfb" version="11"
description="Call Forwarding Busy" active="false" enabled="false">
                                <msd>
                                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
                                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                <item value="cfu"/>
                                <item value="assistant"/>
                                <item value="follow_me"/>
                                <item value="tsmn_request"/>
                                </property_list>
                                </msd>
                                </ssd>
                                <property name="forwarding_ss"
type="boolean" description="" value="true"/>
                                </ssd>
                                <csd>
                                <property name="number" type="address"
                                <property name="external_number"
type="address" description="External forward number" value=""/>
                                <property name="use_external_number"
                                <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
                                </csd>
                                </src_ss>
                                <src_ss src="ss_cf_sip.xml"
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cf_sip" version="9"
description="Call Forwarding by SIP redirection (302 message)" active="false" enabled="false">
                                <msd>
                                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>

```

```

                                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                <item value="cfu"/>
                                <item value="assistant"/>
                                <item value="follow_me"/>
                                <item value="tsmn_request"/>
                                </property_list>
                                </msd>
                                <ssd>
                                <property name="forwarding_ss"
type="boolean" description="" value="true"/>
                                </ssd>
                                </src_ss>
                                <src_ss src="ss_cf_aon.xml"
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cf_aon" version="9"
                                description="Call Forwarding AON" active="false" enabled="false">
                                <msd>
                                <property name="owner" type="atom"
                                description="SS type. Available values: user | trunk | any" value="any"/>
                                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                <item value="assistant"/>
                                <item value="follow_me"/>
                                <item value="cfu_type2"/>
                                <item value="meet_me_conference"/>
                                <item value="cfu"/>
                                <item value="find_me"/>
                                </property_list>
                                </msd>
                                <ssd>
                                <property name="forwarding_ss"
type="boolean" description="" value="true"/>
                                </ssd>
                                <csd>
                                <property name="no_reply"
type="boolean" description="Activate service when subscriber no reply" value="false"/>
                                <property name="timeout"
type="positive_integer" description="Timeout before call forward (sec)" value="15"/>
                                <property name="number" type="address"
                                description="Forward number"/>
                                <property_list name="prefix_number"
type="address" maxlen="10" description="Prefix number A">
                                <undefined/>
                                </property_list>
                                </csd>
                                </src_ss>
                                <src_ss src="ss_cc_agent.xml"

```

```

xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cc_agent" version="2"
description="Contact-Center agent" active="false" enabled="false"/>
    <src_ss src="ss_callback.xml"
        xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="callback" version="5"
description="CallBack" active="false" enabled="false">
        <ssd>
            <property name="collect_digits_time
out" type="integer" description="Wait timeout to collect cancel digits" value="60"/>
            <property name="a_onhook_timeout"
type="positive_integer" description="Timeout for A subscriber onhook (sec)" value="5"/>
            <property name="max_call_retry"
type="positive_integer" description="The maximum count of call's retry to initial subscriber"
value="5"/>
            <property name="dtmf_detector"
type="boolean" description="" value="true"/>
        </ssd>
    </src_ss>
    <src_ss src="ss_call_recording.xml"
        xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="call_recording" version="11"
description="Call recording" active="false" enabled="false">
        <msd>
            <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
            <property name="enumMode"
type="string" description="Internal field to validate mode property" value="always_on|
on_demand|after_answer"/>
        </msd>
        <ssd>
            <property name="locked"
type="boolean" description="Can subscriber manage current SS, or not" value="true"/>
            <property name="dtmf_detector"
type="boolean" description="" value="true"/>
        </ssd>
        <csd>
            <property name="mode" type="enum"
description="The mode of the recording call (always_on|on_demand)"/>
            <property name="separated"
type="boolean" description="The separated recording of the call. If true, the outgoing audio
of the subscriber is recorded to first audio track, the incoming audio of the subscriber is
recorded to second audio track. If false, the incoming and the outgoing audio of the
subscriber is recorded to first audio track." value="false"/>
        </csd>
    </src_ss>
    <src_ss src="ss_boss_group.xml"
        xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="boss_group" version="1"
description="Boss-group" active="false" enabled="false">

```

```

                                <msd>
      <property name="integer_rangeDelay
_ring_timeout" type="string" description="Internal field" value="[0,60]"/>
                                <property name="integer_rangelines
_count" type="string" description="Internal field" value="[0,16]"/>
                                <property_list name="conflict_list
" type="atom" maxlen="0" description="List of SS that is conflict with current SS">
      <item value="cw"/>
                                </property_list>
                                <property_list name="dependency_li
st" type="atom" maxlen="0" description="List of SS that is depended from SS">
      <item value="auto_redial"/>
                                </property_list>
                                </msd>
      <ssid>
                                <property name="dtmf_detector"
type="boolean" description="" value="true"/>
                                </ssid>
      <csd>
                                <property name="display_name"
type="string" description="Display name for boss group" value=""/>
                                <property name="delay_ring_timeout
" type="integer_range" description="Timeout before call on delayed members" value="0"/>
                                <property name="lines_count"
type="integer_range" description="Incoming boss-group lines count" value="2"/>
                                <property name="members_fwd_busy_o
verride" type="boolean" description="Ignore members CFB for group calls" value="true"/>
                                <property name="call_waiting_tone"
type="boolean" description="Play call waiting tone for second(s) calls on group"
value="false"/>
                                <property_list name="members"
type="address" maxlen="8" description="Members for immediate call">
      <undefined/>
                                </property_list>
                                <property_list name="delay_members
" type="address" maxlen="8" description="Members for delay call">
      <undefined/>
                                </property_list>
                                <property_list name="silent_member
s" type="address" maxlen="8" description="Members for BLF and pickup only group calls">
      <undefined/>
                                </property_list>
                                </csd>
      </src_ss>
      <src_ss src="ss_blf.xml"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="blf" version="3"
description="Busy Lamp Field" active="false" enabled="false">
      <csd>

```

```

        <property name="max_subscribe"
type="positive_integer" description="Maximum number of subscriptions" value="16"/>
        <property name="ignore_page_calls"
type="boolean" description="Doesn't send BLF notification from Zone Page calls" value="true"/
>
        </csd>
    </src_ss>
    <src_ss src="ss_avatar.xml"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="avatar" version="2"
description="Picture for outgoing calls" active="false" enabled="false">
        <csd>
            <property name="pic_addr"
type="uri_string" description="URL to access the image" value=""/>
        </csd>
    </src_ss>
    <src_ss src="ss_auto_redial_with_callback.
xml"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="auto_redial_with_callback"
version="6" description="Auto redial with callback" active="false" enabled="false">
        <msd>
            <property name="enumRingback"
type="string" description="Ringback" value="tone|moh"/>
        </msd>
        <csd>
            <property name="recall_timer"
type="positive_integer" description="Time between recalls (sec)" value="90"/>
            <property name="recall_count"
type="positive_integer" description="Recalls' count" value="30"/>
            <property name="a_no_answer_timeou
t" type="positive_integer" description="Awaiting A-side answer timeout (sec)" value="10"/>
            <property name="b_no_answer_timeou
t" type="positive_integer" description="Awaiting B-side answer timeout (sec)" value="30"/>
            <property name="ringback"
type="enum" description="Ringback (tone | moh)" value="tone"/>
        </csd>
    </src_ss>
    <src_ss src="ss_auto_redial.xml"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="auto_redial" version="2"
description="Auto redial" active="false" enabled="false">
        <csd>
            <property name="recall_timeout"
type="positive_integer" description="Timeout between recalls (sec)" value="15"/>
            <property name="recall_count"
type="positive_integer" description="Recalls' count" value="6"/>
            <property name="no_answer_timeout"
type="positive_integer" description="Awaiting answer timeout (sec)" value="30"/>
        </csd>

```

```

        </src_ss>
        <src_ss src="ss_assistant.xml"
            xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="assistant" version="2"
            description="Chief-assistant group" active="false" enabled="false">
            <msd>
                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="user"/>
            </msd>
            <ssd>
                <property name="forwarding_ss"
type="boolean" description="" value="true"/>
            </ssd>
            <csd>
                <property_list name="assistants"
type="index_address" maxlen="4" description="Assistants numbers. Format: <address>[/
<index>]">
                    <undefined/>
                </property_list>
                <property_list name="whitelist"
type="index_address" maxlen="128" description="White list members. Format: <address>[/
<index>]">
                    <undefined/>
                </property_list>
            </csd>
        </src_ss>
        <src_ss src="ss_alarm.xml"
            xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="alarm" version="8"
            description="Alarm Call" active="false" enabled="false">
            <msd>
                <property name="handle_module"
type="atom" description="" value="ss_alarm_handler"/>
            </msd>
            <ssd>
                <property name="collect_digits_tim
eout" type="integer" description="Wait timeout to collect cancel digits" value="60"/>
            </ssd>
            <csd>
                <property name="no_answer_timeout"
type="integer" description="Awaiting answer timer (value in seconds)" value="45"/>
                <property name="call_attempt_count
" type="integer" description="The count of the call's attempt" value="3"/>
                <property name="call_attempt_timeo
ut" type="integer" description="Awaiting timer between calls' attempts" value="60"/>
                <property name="cancel_digits"
type="address" description="Digits to cancel alarm" value="*"/>
                <property name="alarm_1"
type="term" description="" {" " 'one-time' " | daily, [" Day1[, Day2][, ... ] "], {"HH, MM"}[,
AlarmName]}"" value="off"/>

```

```

                                <property name="alarm_2"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="alarm_3"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="alarm_4"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="alarm_5"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="alarm_6"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="alarm_7"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="alarm_8"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="alarm_9"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                </csd>
                                </src_ss>
                                <src_ss src="ss_acb.xml"
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="acb" version="2"
description="Anonymous Call Blocking" active="false" enabled="false"/>
                                <src_ss src="ss_3way.xml"
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="3way" version="6"
description="3-Way Conference" active="false" enabled="false">
                                <msd>
                                <property_list name="dependenc
y_list" type="atom" maxlen="0" description="List of SS that is depended from SS">
                                <item value="chold"/>
                                </property_list>
                                <property_list name="conflict_
list" type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                <item value="conference"/>
                                </property_list>
                                </msd>
                                <ssd>
                                <property name="dtmf_detector"
type="boolean" description="" value="true"/>
                                </ssd>
                                </src_ss>
                                </sss>

```

```
</out>
```

Hc_ss_calls_history

Команда предназначена для вывода списка услуг, используемый в вызове.

Аналог команды в CoCon:

`domain/<DOMAIN>/calls/show_ss`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://&server&port;/commands/Hc_ss_calls_history

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема входного и выходного XML


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="ss_calls_history">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Copyright (C) 2015, Eltex. All right reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <!-- Types -->

  <xs:simpleType name="dateTimeType">
    <xs:restriction base="xs:string">
      <!-- DD.MM.YYYY HH:MM:SS -->
      <xs:pattern value="\d{2}\.\d{2}\.\d{4}\s{1}\d{2}:\d{2}:\d{2}"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="ssType">
    <xs:attribute name="call_ref" type="xs:nonNegativeInteger" use="required"/>
    <xs:attribute name="ss_name" type="xs:string" use="required"/>
    <xs:attribute name="addr_a" type="xs:string" use="required"/>
    <xs:attribute name="addr_b" type="xs:string" use="required"/>
    <xs:attribute name="participants" type="xs:string" use="required"/>
    <xs:attribute name="activate_time" type="dateTimeType" use="required"/>
    <xs:attribute name="activator" type="xs:string" use="required"/>
    <xs:attribute name="action" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- End of types -->

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain" minOccurs="1" maxOccurs="1">
                <xs:complexType>
                  <xs:attribute name="name"
                    type="xs:string"
                    use="required"/>
                  <xs:attribute name="call_ref"
                    type="xs:nonNegativeInteger"
                    use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

        </xs:all>
        </xs:complexType>
        </xs:element>
    </xs:all>
</xs:complexType>
</xs:element>
<!-- /in -->

<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="response">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="ss" type="ssType"
                            minOccurs="0" maxOccurs="unbounded"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
<!-- /out -->

</xs:schema>

```

Запрос: http://192.168.1.21:9999/commands/ss_calls_history

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_calls_history.xsd">
  <request>
    <domain name="biysk.local" call_ref="742531227"/>
  </request>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_calls_history.xsd">
  <response/>
</out>
```

Hc_ss_deactivate

Команда позволяет деактивировать услугу в определенном домене для определенного абонента (либо для всех абонентов, если ранее услуга было активировано для всех абонентов).

Аналог команды в CoCon:

```
/cluster/storage/<CLUSTER>/ss/deactivate
/domain/<DOMAIN>/ss/deactivate
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_deactivate

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="address" type="xs:string" use="optional"/>
    <xs:attribute name="ss" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="address" type="xs:string" use="optional"/>
    <xs:attribute name="ss" type="xs:string" use="optional"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- ssDeactivateType -->
  <xs:complexType name="ssDeactivateType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ss" type="ssDeactivateType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok" type="okType" minOccurs="0"
maxOccurs="unbounded"/>
              <xs:element name="error" type="errorType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

        </xs:element>
    </xs:all>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример :

Деактивировать услугу **cfu** у абонента с номером 240466 в домене biysk.local .

Запрос:

http://192.168.1.21:9999/commands/ss_deactivate

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_deactivate.xsd">
  <request domain="biysk.local">
    <ss address="240466" name="cfu" />
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_deactivate.xsd">
  <response domain="biysk.local">
    <ok address="240466" ss="cfu"/>
  </response>
</out>

```

Ис_ss_enable

Команда включает/выключает услугу указанным абонентам.

Аналог команд в CoCon:

/domain/<DOMAIN>/ss/enable

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_enable

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <!-- ssEnableType -->
  <xs:complexType name="ssEnableType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="enable" type="xs:boolean" use="required"/>
  </xs:complexType>

  <!-- okType -->
  <xs:complexType name="okType">
    <xs:attribute name="ss" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- errorType -->
  <xs:complexType name="errorType">
    <xs:attribute name="ss" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="ssEnableListType">
    <xs:sequence>
      <xs:element name="ss" type="ssEnableType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="addressListType">
    <xs:sequence>
      <xs:element name="address" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="address_list" type="addressListType"/>
              <xs:element name="ss_list" type="ssEnableListType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

        <xs:attribute name="domain" type="xs:string" use="required"/>
    </xs:complexType>
</xs:element>
</xs:all>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:all>
            <xs:element name="response">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="ok" type="okType"/>
                        <xs:element name="error" type="errorType"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
        </xs:all>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/ss_enable

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local">
    <address_list>
      <address name="240464"/>
      <address name="240465"/>
    </address_list>
    <ss_list>
      <ss name="m1" enable="true"/>
      <ss name="conference" enable="false"/>
    </ss_list>
  </request>
</in>

```

Ответ: 200


```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_enable.xsd">
  <response>
    <ok ss="conference" address="240465"/>
    <ok ss="conference" address="240464"/>
    <error address="240465" ss="m1" reason="Address 240465 doesn't access to supplementary
service m1"/>
    <error address="240464" ss="m1" reason="Address 240464 doesn't access to supplementary
service m1"/>
  </response>
</out>
```

Ис_ss_enabled

Команда просмотра включенных услуг для указанных пользователей.

Аналог команд в CoCon:

```
/cluster/storage/<CLUSTER>/ss/enabled
/domain/<DOMAIN>/ss/enabled
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_enabled

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:complexType name="enabledSS">
    <xs:attribute name="name"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="address" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ss" type="enabledSS" minOccurs="1"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/ss_enabled

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
  <request domain="biysk.local" address="240465"/>
</in>

```

Ответ:200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_enabled.xsd">
  <response>
    <ss name="3way"/>
    <ss name="acb"/>
    <ss name="alarm"/>
    <ss name="assistant"/>
    <ss name="auto_redial"/>
    <ss name="auto_redial_with_callback"/>
    <ss name="avatar"/>
    <ss name="blf"/>
    <ss name="boss_group"/>
    <ss name="call_recording"/>
    <ss name="callback"/>
    <ss name="cc_agent"/>
    <ss name="cf_aon"/>
    <ss name="cf_sip"/>
    <ss name="cfb"/>
    <ss name="cfnr"/>
    <ss name="cfnr_type2"/>
    <ss name="cfos"/>
    <ss name="cft"/>
    <ss name="cfu"/>
    <ss name="cfu_type2"/>
    <ss name="cgg"/>
    <ss name="chold"/>
    <ss name="chunt"/>
    <ss name="click_to_call"/>
    <ss name="clip"/>
    <ss name="clir"/>
    <ss name="cliro"/>
    <ss name="conference"/>
    <ss name="cpip"/>
    <ss name="ctr"/>
    <ss name="cw"/>
    <ss name="direct_call"/>
    <ss name="distinctive_picture"/>
    <ss name="distinctive_ring"/>
    <ss name="dnd"/>
    <ss name="fax_receiver"/>
    <ss name="fax_to_email"/>
    <ss name="fbc"/>
    <ss name="find_me"/>
    <ss name="find_me_no_response"/>
    <ss name="flexicall"/>
    <ss name="follow_me"/>
    <ss name="follow_me_no_response"/>
```

```

<ss name="intercom"/>
<ss name="intervention"/>
<ss name="mclid"/>
<ss name="meet_me_conference"/>
<ss name="message"/>
<ss name="ml"/>
<ss name="multicast_listen"/>
<ss name="multicast_page"/>
<ss name="my_number"/>
<ss name="park"/>
<ss name="personal_ivr"/>
<ss name="pickup"/>
<ss name="presence"/>
<ss name="qsig_cfb"/>
<ss name="qsig_cfnr"/>
<ss name="qsig_cfu"/>
<ss name="rbp"/>
<ss name="redial"/>
<ss name="remote_phone"/>
<ss name="rfc"/>
<ss name="ring_back_tone"/>
<ss name="sca"/>
<ss name="sco_black"/>
<ss name="sco_white"/>
<ss name="scr"/>
<ss name="speed_dial"/>
<ss name="teleconference"/>
<ss name="teleconference_manager"/>
<ss name="time_service"/>
<ss name="tsmn_request"/>
<ss name="vip_call"/>
<ss name="voice_page"/>
<ss name="voice_page_control"/>
<ss name="voicemail"/>
<ss name="walkie_talkie"/>
<ss name="zone_page"/>
</response>
</out>

```

Ис_ss_info

Команда позволяет вывести список активированных услуг в рамках определенного домена.

Аналог команд в CoCon:

```

/cluster/storage/<CLUSTER>/ss/info*
/domain/<DOMAIN>/ss/info*

```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом:

ss_info.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ss.xsd"/>

  <xs:complexType name="ssInfoType">
    <xs:complexContent>
      <xs:extension base="ssType">
        <xs:attribute name="domain" type="xs:string" use="required"/>
        <xs:attribute name="address" type="xs:string" use="required"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- ssListType -->
  <xs:complexType name="ssInfoListType">
    <xs:sequence>
      <xs:element name="ss" type="ssInfoType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="address" type="xs:string" use="optional"/>
            <xs:attribute name="ss_name" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="sss" type="ssInfoListType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Схема данных описана в разделе [Hc_ss - Управление услугами](#).

Пример 1:

Запросить информацию по всем ДВО для определенного адреса:

Запрос: http://192.168.1.21:9999/commands/ss_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_info.xsd">
  <request domain="biysk.local" address="240101"/>
</in>
```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_info.xsd">
  <sss>
    <ss domain="biysk.local" address="240101"
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="zone_page" version="10" description="Zone Page
      call" active="false" enabled="true">
      <msd>
        <property name="enumMulticast_codec" type="string" description="Internal
        field" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|G726-40"/>
        <property name="integer_rangeMulticast_ttl" type="string"
        description="Internal field" value="[1,255]"/>
        <property name="integer_rangeMulticast_port" type="string"
        description="Internal field" value="[0,65535]"/>
        <property name="enumInitiation_policy" type="string" description="Initiation
        call's policy" value="managers|members|any"/>
        <property name="enumPriority" type="string" description="Call's priority"
        value="0_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
      </msd>
      <ssd>
        <property name="dtmf_detector" type="boolean" description="" value="true"/>
      </ssd>
      <csd>
        <property name="multicast_ttl" type="integer_range" description="TTL for
        multicast" value="1"/>
        <property name="multicast_codec" type="enum" description="Codec to be used for
        multicasting" value="PCMA"/>
        <property name="multicast_port" type="integer_range" description="Port of
        multicast address" value="0"/>
        <property name="multicast_ip" type="ip" description="Optional multicast
        address to use for announcement" value="0.0.0.0"/>
        <property name="announcement_repeat_digit" type="address" description="Digit
        to repeat announcement files" value="#" />
        <property name="play_release_tone" type="boolean" description="ZonePage
        release tone" value="false"/>
        <property name="display_name" type="string" description="ZonePage call
        DisplayName" value="ZonePage"/>
        <property name="initiation_policy" type="enum" description="Initiation call's
        policy" value="any"/>
        <property name="terminate_when_last_listener_leave" type="boolean"
        description="Terminate call after last listener disconnects"/>
        <property name="terminate_when_playback_ends" type="boolean"
        description="Terminate call after playback ends"/>
        <property name="terminate_when_initiator_disconnects" type="boolean"
        description="Terminate call after initiator disconnected"/>
        <property name="announcement_repeat_time" type="integer"
        description="Announcement files repeat timeout" value="10"/>

```



```

        <property name="announcement_playback_cnt" type="integer"
description="Announcement files playback count" value="1"/>
        <property name="announcement_pause" type="integer" description="Pause before
play announcement files" value="0"/>
        <property name="intro_pause" type="integer" description="Pause before play
intro files" value="3"/>
        <property name="priority" type="enum" description="(0_flash_override | 1_flash
| 2_immediate | 3_priority | 4_routine)" value="3_priority"/>
        <property_list name="announcement_files" type="media_resource" maxlen="10"
description="List of Announcement media_files"/>
        <property_list name="intro_files" type="media_resource" maxlen="10"
description="List of Intro media_files"/>
        <property_list name="managers" type="address" maxlen="256" description="Zone
page managers list"/>
        <property_list name="members" type="address" maxlen="256" description="Zone
page member list">
            <undefined/>
        </property_list>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="walkie_talkie" version="1" description="Walkie
Talkie" active="false" enabled="true">
    <ssd>
        <property name="dtmf_detector" type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="swap_digit" type="address" description="Digit to swap
reception / transmission mode" value="*/>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="voicemail" version="13" description="Voice mail
service" active="false" enabled="true">
    <msd>
        <property_list name="conflict_list" type="atom" maxlen="0" description="List
of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
            <item value="cfu"/>
            <item value="find_me"/>
            <item value="cf_aon"/>
            <item value="cft"/>
        </property_list>
    </msd>
</ssd>

```

```

        <property name="dtmf_detector" type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="dnd" type="boolean" description="Activate service when
subscriber activate DND" value="false"/>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="voice_page_control" version="2"
description="'Voice Page Control' service" active="false" enabled="true">
    <csd>
        <property name="mute_incoming_vp" type="boolean" description="Accept incoming
voice page call in silent mode" value="true"/>
        <property name="block_incoming_vp_by_dnd" type="boolean" description="Block
voice page calls by DND service" value="true"/>
        <property name="accept_incoming_vp" type="boolean" description="Accept or
Reject voice page calls" value="true"/>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="voice_page" version="9" description="Voice Page
call" active="false" enabled="true">
    <msd>
        <property name="enumPriority" type="string" description="Call's priority"
value="0_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
    </msd>
    <ssd>
        <property name="dtmf_detector" type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="announcement_repeat_digit" type="address" description="Digit
to repeat announcement files" value="#" />
        <property name="swap_digit" type="address" description="Digit to swap
reception / transmission mode" value="*" />
        <property name="announcement_playback_cnt" type="integer"
description="Announcement files playback count [0..5]" value="0"/>
        <property name="announcement_repeat_time" type="integer"
description="Announcement files repeat timeout" value="10"/>
        <property name="priority" type="enum" description="(0_flash_override | 1_flash
| 2_immediate | 3_priority | 4_routine)" value="3_priority"/>
        <property name="skip_cf" type="boolean" description="Skipping call forwarding"
value="true"/>
        <property_list name="announcement_files" type="media_resource" maxlen="10"
description="List of Announcement media_files"/>
        <property_list name="intro_files" type="media_resource" maxlen="10"
description="List of Intro media_files"/>
    </csd>
</ss>

```

```

<ss domain="biysk.local" address="240101"
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="vip_call" version="4" description="Vip call"
  active="false" enabled="true">
  <msd>
    <property name="enumPriority" type="string" description="Call's priority"
value="0_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
  </msd>
  <csd>
    <property name="skip_low_priority_call" type="boolean" description="Skipping
second call if he has with low priority." value="false"/>
    <property name="implicit_mode" type="boolean" description="Use vip call
without feature code." value="false"/>
    <property name="priority" type="enum" description="(0_flash_override | 1_flash
| 2_immediate | 3_priority | 4_routine)" value="4_routine"/>
  </csd>
</ss>
<ss domain="biysk.local" address="240101"
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="tsmn_request" version="3" description="Support
request on TSMN" active="false" enabled="true">
  <msd>
    <property name="owner" type="atom" description="SS type. Available values:
user | trunk | any" value="trunk"/>
  </msd>
  <csd>
    <property name="tsmn_trunk_backup" type="trunk" description="Backup trunk to
TSMN server" value=""/>
    <property name="tsmn_trunk" type="trunk" description="Trunk to TSMN server"/>
    <property name="timeout" type="positive_integer" description="Timeout on TSMN
request(s) (msec)" value="3000"/>
  </csd>
</ss>
<ss domain="biysk.local" address="240101"
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="teleconference_manager" version="3"
description="Teleconference manager" active="true" enabled="true">
  <csd>
    <property_list name="second_line" type="address" maxlen="10"
description="Second line numbers"/>
  </csd>
</ss>
<ss domain="biysk.local" address="240101"
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="teleconference" version="4"
description="Teleconference" active="true" enabled="true">
  <msd>
    <property name="enumType" type="string" description="Teleconference type:
meeting or room" value="meeting|room"/>
  </msd>

```

```

    <csd>
      <property name="destroy_by_no_initiator" type="boolean" description="Destroy
the teleconference by release from the initiator?" value="true"/>
      <property name="type" type="enum" description="Teleconference type: meeting or
room" value="meeting"/>
      <property name="template_from_room" type="teleconference_room"
description="Get template from the teleconference room" value=""/>
    </csd>
  </ss>
  <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="speed_dial" version="5" description="Speed dial
(Abbreviated dialing)" active="false" enabled="true">
    <msd>
      <property name="enumType" type="string" description="The type of address book.
address_book_10 supports 10 number in the address book. address_book_100 supports 100 numbers
on the address book" value="address_book_10|address_book_100"/>
    </msd>
    <csd>
      <property name="type" type="enum" description="(address_book_10 |
address_book_100)" value="address_book_100"/>
      <property_list name="map" type="speed_dial_item" maxlen="50" description="Map
between short dial and full address. Format: <short_dial>/<address>"/>
    </csd>
  </ss>
  <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="smart_cancel" version="2" description="Smart
cancel service" active="false" enabled="true">
    <msd>
      <property name="integer_rangeTimeout" type="string" description="Timeout range
property" value="[0,5]"/>
    </msd>
    <csd>
      <property name="timeout" type="integer_range" description="Timeout to wait
release" value="2"/>
    </csd>
  </ss>
  <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="sip_message" version="1" description="SIP Message
service" active="false" enabled="true">
    <msd>
      <property name="enumMode" type="string" description="Call's priority"
value="sendonly|recvonly|sendrecv"/>
    </msd>
    <csd>
      <property name="mode" type="enum" description="(sendonly | recvonly |
sendrecv)" value="sendrecv"/>
    </csd>
  </ss>

```

```

</ss>
<ss domain="biysk.local" address="240101"
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="second_handset" version="1" description="Second
  Handset" active="false" enabled="true"/>
  <ss domain="biysk.local" address="240101"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="scr" version="2" description="Selective Call
    Rejection" active="false" enabled="true">
    <csd>
      <property_list name="blacklist" type="index_address" maxlen="50"
        description="Black list members. Format: <address>[/<index>]">
        <undefined/>
      </property_list>
    </csd>
  </ss>
  <ss domain="biysk.local" address="240101"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="sco_white" version="2" description="Selective
    Call Originating, White list" active="false" enabled="true">
    <csd>
      <property_list name="whitelist" type="index_address" maxlen="50"
        description="White list members. Format: <address>[/<index>]">
        <undefined/>
      </property_list>
    </csd>
  </ss>
  <ss domain="biysk.local" address="240101"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="sco_black" version="2" description="Selective
    Call Originating, Black list" active="false" enabled="true">
    <csd>
      <property_list name="blacklist" type="index_address" maxlen="50"
        description="Black list members. Format: <address>[/<index>]">
        <undefined/>
      </property_list>
    </csd>
  </ss>
  <ss domain="biysk.local" address="240101"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="sca" version="2" description="Selective Call
    Acceptance" active="false" enabled="true">
    <csd>
      <property_list name="whitelist" type="index_address" maxlen="50"
        description="White list members. Format: <address>[/<index>]">
        <undefined/>
      </property_list>
    </csd>
  </ss>
  <ss domain="biysk.local" address="240101"

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="ring_back_tone" version="2" description="Ring
back tone" active="false" enabled="true">
        <csd>
            <property name="file" type="media_resource" description="File to play on
ring"/>
        </csd>
    </ss>
    <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="rfc" version="2" description="Rejection of
Forwarded Calls" active="false" enabled="true"/>
        <ss domain="biysk.local" address="240101"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="remote_ring" version="1" description="Remote ring
to callee" active="false" enabled="true">
            <msd>
                <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
                    <item value="distinctive_ring"/>
                </property_list>
            </msd>
            <csd>
                <property_list name="announcement_files" type="media_resource"
maxlen="10" description="List of Announcement media_files">
                    <item value="domain://sounds/7820.wav"/>
                </property_list>
            </csd>
        </ss>
        <ss domain="biysk.local" address="240101"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="remote_phone" version="1" description="Remote
phone" active="false" enabled="true">
            <msd>
                <property name="handle_module" type="atom" description=""
value="ss_remote_phone_handler"/>
            </msd>
            <csd>
                <property name="pin" type="address" description="PIN-code"/>
                <property name="remote_endpoint" type="interface" description="Remote
interface" value=""/>
            </csd>
        </ss>
        <ss domain="biysk.local" address="240101"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="redial" version="2" description="Last number
redial" active="false" enabled="true"/>
            <ss domain="biysk.local" address="240101"

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="rbp" version="3" description="SS for restricting
        call-processes by NI and PIN" active="false" enabled="true">
        <csd>
            <property name="implicit_call" type="boolean" description="Use
            implicit RBP call (Request PIN-code in case of use recsricted call)" value="false"/>
            <property_list name="restricted_ni" type="ni" maxlen="6"
            description="Restricted number indication">
                <undefined/>
            </property_list>
        </csd>
    </ss>
    <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfu" version="3" description="QSIG: Call
        Forwarding Unconditional" active="false" enabled="true">
        <msd>
            <property name="owner" type="atom" description="SS type. Available
            values: user | trunk | any" value="trunk"/>
            <property_list name="conflict_list" type="atom" maxlen="0"
            description="List of SS that is conflict with current SS">
                <item value="cfu"/>
                <item value="follow_me"/>
            </property_list>
        </msd>
        <csd>
            <property name="hide_forwarding" type="boolean" description="Hide
            forwarding fields (rgn, ocdrn)" value="false"/>
        </csd>
    </ss>
    <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfnr" version="2" description="QSIG: Call
        Forwarding No Reply" active="false" enabled="true">
        <msd>
            <property name="owner" type="atom" description="SS type. Available
            values: user | trunk | any" value="trunk"/>
            <property_list name="conflict_list" type="atom" maxlen="0"
            description="List of SS that is conflict with current SS">
                <item value="cfu"/>
            </property_list>
        </msd>
    </ss>
    <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfb" version="3" description="QSIG: Call
        Forwarding Busy" active="false" enabled="true">
        <msd>
            <property name="owner" type="atom" description="SS type. Available
            values: user | trunk | any" value="trunk"/>

```

```

        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="cfu"/>
        </property_list>
    </msd>
    <csd>
        <property name="hide_forwarding" type="boolean" description="Hide
forwarding fields (rgn, ocdrn)" value="false"/>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="privacy" version="2" description="Privacy"
active="false" enabled="true">
    <msd>
        <property name="enumMode" type="string" description="Mode"
value="on_demand|always_on"/>
    </msd>
    <ssd>
        <property name="dtmf_detector" type="boolean" description=""
value="true"/>
    </ssd>
    <csd>
        <property name="mode" type="enum" description="Privacy usage mode"
value="on_demand"/>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="presence" version="2" description="SIP Presence"
active="false" enabled="true">
    <csd>
        <property name="max_subscribe" type="positive_integer"
description="Maximum number of subscriptions" value="16"/>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="pickup" version="2" description="Call Pickup"
active="true" enabled="true">
    <csd>
        <property_list name="pickup_groups" type="index_string"
maxlen="10" description="Call pick-up's groups names. Format: <Name>[/<Inx>]">
            <item value="gr1"/>
            <item value="gr2"/>
            <item value="333gr"/>
        </property_list>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"

```



```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="personal_ivr" version="2" description="Personal
IVR script" active="true" enabled="true">
        <msd>
            <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="user"/>
            <property name="handle_module" type="atom" description=""
value="ss_personal_ivr_handler"/>
        </msd>
        <csd>
            <property name="script" type="ivr_script_id" description="Script
ID" value=""/>
        </csd>
    </ss>
    <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="park" version="2" description="Call park"
active="true" enabled="true"/>
        <ss domain="biysk.local" address="240101"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="multicast_page" version="1"
description="Multicast IP Paging" active="false" enabled="true">
            <msd>
                <property name="integer_rangeTtl" type="string"
description="Internal field" value="[1,255]"/>
                <property name="integer_rangePort" type="string"
description="Internal field" value="[10000,65535]"/>
                <property name="enumCodec" type="string" description="Internal
field" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|G726-40"/>
            </msd>
            <csd>
                <property name="ttl" type="integer_range" description="TTL for
multicast packets (1-255)" value="1"/>
                <property name="codec" type="enum" description="Codec to be
used for multicasting" value="PCMA"/>
                <property name="port" type="integer_range" description="UDP
port to cast the media to (10000-65535)"/>
                <property name="ip" type="ip" description="IP address to cast
the media to"/>
            </csd>
        </ss>
        <ss domain="biysk.local" address="240101"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="multicast_listen" version="1"
description="Multicast Listen Service" active="false" enabled="true">
            <msd>
                <property name="integer_rangePort" type="string"
description="Internal field to udp port property" value="[10000,65535]"/>

```

```

        <property name="enumCodec" type="string" description="Internal
field to codec property" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|
G726-40"/>
    </msd>
    <csd>
        <property name="codec" type="enum" description="Audio codec
for multicast traffic." value="G722"/>
        <property name="port" type="integer_range" description="Port
for listen multicast (10000-65535)."/>
        <property name="ip" type="ip" description="Ip address for
listen multicast."/>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="m1" version="4" description="Multiline"
active="true" enabled="true">
    <msd>
        <property name="integer_rangeLine_count_out" type="string"
description="Internal field" value="[-1,16]"/>
        <property name="integer_rangeLine_count" type="string"
description="Internal field" value="[0,16]"/>
    </msd>
    <csd>
        <property name="line_count_out" type="integer_range"
description="Outgoing lines count" value="2"/>
        <property name="line_count" type="integer_range"
description="Line count" value="2"/>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="message" version="2" description="Message
service" active="false" enabled="true">
    <msd>
        <property name="owner" type="atom" description="SS type.
Available values: user | trunk | any" value="any"/>
    </msd>
    <ssd>
        <property name="remove_only_after_answer" type="boolean"
description="Remove message only after abonent answer" value="false"/>
        <property name="message_box_size" type="integer"
description="Message box size" value="20"/>
    </ssd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="meet_me_conference" version="2" description="Meet
Me Conference" active="false" enabled="true">
    <msd>

```

```

        <property name="enumDestroy_mode" type="string"
description="Internal field to validate mode property" value="by_no_master|by_no_more_calls"/>
        <property name="regimeMode" type="string"
description="Internal field to validate mode property" value="master:destroy_mode:room_number|
member:room_number"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
        </property_list>
    </msd>
</csd>
    <property name="room_number" type="address" description="Room
number" value=""/>
    <property name="destroy_mode" type="enum" description="The
mode of destroying conference (by_no_master|by_no_more_calls)" value="by_no_more_calls"/>
    <property name="mode" type="regime" description="The mode of
meet me conference (master|member)"/>
</csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="mcid" version="2" description="Malicious Call
Identification" active="true" enabled="true"/>
    <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="introduce" version="1" description="Introduce
announcement from callee" active="false" enabled="true">
        <csd>
            <property name="announcement" type="media_resource"
description="Announcement media file"/>
        </csd>
    </ss>
    <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="intervention" version="4"
description="Intervention" active="true" enabled="true">
        <msd>
            <property name="enumRestriction" type="string"
description="Internal field to validate mode property" value="full_control|
implicitly_allowed|explicitly_allowed"/>
        </msd>
        <ssd>
            <property name="dtmf_detector" type="boolean"
description="" value="true"/>
        </ssd>
    </csd>
    <property name="restriction" type="enum" description="The
restriction to the intervention call (full_control|allowed_only)" value="full_control"/>

```

```

        </csd>
    </ss>
    <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="intercom" version="1" description="Intercom"
        active="false" enabled="true"/>
        <ss domain="biysk.local" address="240101"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="hide_cf_name" version="1" description="Hide call
forwarding name for subscriber A" active="false" enabled="true"/>
            <ss domain="biysk.local" address="240101"
                xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
                xs:noNamespaceSchemaLocation="ss.xsd" name="fxo_trunk" version="1" description="FXO trunk
service" active="false" enabled="true">
                <msd>
                    <property name="owner" type="atom" description="SS
type. Available values: user | trunk | any" value="trunk"/>
                </msd>
            </ss>
            <ss domain="biysk.local" address="240101"
                xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
                xs:noNamespaceSchemaLocation="ss.xsd" name="follow_me_no_response" version="8"
                description="Follow me no response" active="false" enabled="true">
                <msd>
                    <property_list name="conflict_list" type="atom"
                    maxlen="0" description="List of SS that is conflict with current SS">
                        <item value="cfu"/>
                        <item value="assistant"/>
                        <item value="cft"/>
                        <item value="cgg"/>
                        <item value="follow_me"/>
                    </property_list>
                </msd>
                <ssd>
                    <property name="forwarding_ss" type="boolean"
                    description="" value="true"/>
                </ssd>
            </ss>
        <csd>
            <property name="use_external_number"
            type="boolean" description="Use external forward number" value="false"/>
            <property name="external_number" type="address"
            description="External forward number" value=""/>
            <property name="pin" type="address"
            description="PIN-code" value=""/>
            <property name="timeout" type="positive_integer"
            description="Timeout before call forward (sec)" value="20"/>
            <property name="number" type="address"
            description="Followed number" value=""/>
        </csd>
    </ss>

```

```

        <ss domain="biysk.local" address="240101"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="follow_me" version="11" description="Follow me"
            active="false" enabled="true">
            <msd>
                <property_list name="conflict_list" type="atom"
                    maxlen="0" description="List of SS that is conflict with current SS">
                    <item value="assistant"/>
                </property_list>
            </msd>
            <ssd>
                <property name="forwarding_ss" type="boolean"
                    description="" value="true"/>
            </ssd>
            <csd>
                <property name="use_external_number"
                    type="boolean" description="Use external forward number" value="false"/>
                <property name="external_number" type="address"
                    description="External forward number" value=""/>
                <property name="pin" type="address"
                    description="PIN-code" value=""/>
                <property name="number" type="address"
                    description="Followed number" value=""/>
            </csd>
        </ss>
        <ss domain="biysk.local" address="240101"
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="flexicall" version="1" description="FlexiCall"
            active="false" enabled="true">
            <msd>
                <property name="enumMode" type="string"
                    description="Internal field to validate mode property" value="all|external|internal"/>
                <property name="owner" type="atom" description="SS
                    type. Available values: user | trunk | any." value="user"/>
                <property_list name="conflict_list" type="atom"
                    maxlen="0" description="List of SS that is conflict with current SS">
                    <item value="cgg"/>
                </property_list>
            </msd>
            <ssd>
                <property name="forwarding_ss" type="boolean"
                    description="" value="true"/>
            </ssd>
            <csd>
                <property name="mode" type="enum"
                    description="Mode operating. Available values (all | external | internal)"/>
                <property_list name="internal_numbers"
                    type="address" maxlen="10" description="Group's internal numbers"/>
                <property_list name="external_numbers"
                    type="address" maxlen="10" description="Group's external numbers"/>
            </csd>
        </ss>

```

```

        </csd>
    </ss>
    <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="find_me_no_response" version="6"
        description="Find Me no response" active="false" enabled="true">
        <msd>
            <property_list name="conflict_list" type="atom"
                maxlen="0" description="List of SS that is conflict with current SS">
                <item value="cfu"/>
                <item value="assistant"/>
                <item value="ctu"/>
                <item value="cgg"/>
                <item value="follow_me"/>
                <item value="follow_me_no_response"/>
                <item value="find_me"/>
            </property_list>
        </msd>
        <ssd>
            <property name="forwarding_ss" type="boolean"
                description="" value="false"/>
        </ssd>
        <csd>
            <property name="timeout" type="positive_integer"
                description="Timeout before call forward (sec)" value="20"/>
            <property name="send_call_diversion"
                type="boolean" description="Send call diversion flag in call signalisation" value="true"/>
            <property_list name="groups" type="find_me_group"
                maxlen="32" description="Find me groups"/>
        </csd>
    </ss>
    <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="find_me" version="7" description="Find Me"
        active="false" enabled="true">
        <msd>
            <property_list name="conflict_list" type="atom"
                maxlen="0" description="List of SS that is conflict with current SS">
                <item value="assistant"/>
                <item value="follow_me"/>
                <item value="cfu_type2"/>
                <item value="meet_me_conference"/>
                <item value="cfu"/>
            </property_list>
        </msd>
        <ssd>
            <property name="forwarding_ss" type="boolean"
                description="" value="true"/>
        </ssd>
        <csd>

```

```

        <property name="send_call_diversion"
type="boolean" description="Send call diversion flag in call signalisation" value="true"/>
        <property_list name="groups" type="find_me_group"
maxlen="32" description="Find me groups"/>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="fbc" version="1" description="Forwarding Barring
Call" active="false" enabled="true">
    <msd>
        <property name="owner" type="atom" description="SS
type. Available values: user | trunk | any" value="any"/>
    </msd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="fax_to_email" version="2" description="Fax to
email" active="false" enabled="true">
    <csd>
        <property name="send_error_report" type="boolean"
description="Send email in case of fax received failed" value="false"/>
        <property name="email" type="string"
description="Email address" value=""/>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="fax_receiver" version="2" description="Enable fax
receiving ability for terminals which is not support fax by their own" active="false"
enabled="true"/>
    <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="dnd" version="4" description="Do not
Disturb" active="false" enabled="true">
        <msd>
            <property name="enumMode" type="string"
description="Internal field to validate mode property" value="default|smart"/>
        </msd>
        <csd>
            <property name="mode" type="enum"
description="The mode of the working dnd service (default|smart)" value="default"/>
        </csd>
    </ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="distinctive_ring" version="4"
description="Destination ringtone change" active="false" enabled="true">
    <csd>

```

```

                <property_list name="filters"
type="distinctive_ring" maxlen="16" description="Distinctive ring for specific numbers"/>
                </csd>
            </ss>
            <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="distinctive_picture" version="2"
description="Display caller image by groups" active="false" enabled="true">
                <csd>
                    <property_list name="groups"
type="distinctive_picture" maxlen="20" description="Distinctive picture for specific numbers"/
>
                </csd>
            </ss>
            <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="direct_call" version="2"
description="Direct call" active="false" enabled="true">
                <csd>
                    <property name="timeout"
type="positive_integer" description="Timeout before direct call (sec)" value="5"/>
                    <property name="number" type="address"
description="Call number to direct call"/>
                </csd>
            </ss>
            <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cw" version="5" description="Call
Waiting" active="false" enabled="true">
                <msd>
                    <property_list name="dependency_list"
type="atom" maxlen="0" description="List of SS that is depended from SS">
                        <item value="chold"/>
                    </property_list>
                </msd>
                <ssd>
                    <property name="dtmf_detector" type="boolean"
description="" value="true"/>
                </ssd>
            </ss>
            <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="ctr" version="5" description="Call
Transfer" active="true" enabled="true">
                <msd>
                    <property_list name="dependency_list"
type="atom" maxlen="0" description="List of SS that is depended from SS">
                        <item value="chold"/>
                    </property_list>
                </msd>

```



```

                <ssd>
                    <property name="dtmf_detector" type="boolean"
description="" value="true"/>
                </ssd>
            </ss>
            <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cpip" version="2" description="Calling
Picture Identification Presentation" active="false" enabled="true">
                <msd>
                    <property name="enumDisplay_mode"
type="string" description="Internal field to validate 'display-mode' property" value="icon|
fullscreen"/>
                </msd>
                <csd>
                    <property name="display_mode" type="enum"
description="Sets the picture display mode: icon mode or fullscreen mode." value="icon"/>
                </csd>
            </ss>
            <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="conference" version="9"
description="Conference Call, Add-on" active="true" enabled="true">
                <msd>
                    <property name="enumDestroy_mode"
type="string" description="Internal field to validate mode property" value="by_no_master|
by_no_more_calls"/>
                    <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                        <item value="3way"/>
                    </property_list>
                    <property_list name="dependency_list"
type="atom" maxlen="0" description="List of SS that is depended from SS">
                        <item value="chold"/>
                        <item value="ctr"/>
                    </property_list>
                </msd>
                <csd>
                    <property name="max_participants"
type="positive_integer" description="The maximum number of participants in the conference,
limited to the number 64 at the top. By default 16" value="16"/>
                    <property name="destroy_mode" type="enum"
description="The mode of destroying conference (by_no_master|by_no_more_calls)"
value="by_no_master"/>
                </csd>
            </ss>
            <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cnip" version="3" description="Calling
Name Identification Presentation" active="true" enabled="true">

```

```

        <msd>
            <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
        </msd>
    </ss>
    <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cliro" version="2" description="Calling
Line Identification Restriction Override" active="false" enabled="true"/>
        <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="clir" version="3" description="Calling
Line Identification Restriction" active="false" enabled="true">
            <msd>
                <property name="enumMode" type="string"
description="Internal field to validate mode property" value="user_defined|restrict_all"/>
            </msd>
            <csd>
                <property name="mode" type="enum"
description="The mode of the working clir service (user_defined|restrict_all)"
value="restrict_all"/>
            </csd>
        </ss>
        <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="clip" version="5" description="Calling
Line Identification Presentation" active="true" enabled="true"/>
        <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="click_to_call" version="5"
description="Click to call" active="true" enabled="true">
            <csd>
                <property name="auto_answer"
type="boolean" description="Auto answer" value="true"/>
                <property name="number" type="address"
description="User's number" value="240101"/>
                <property_list name="hashes"
type="string" maxlen="10" description="User's hashes">
                    <item value="bW96aWxsYS81LjAgKHgXM
TsgdWJ1bnR10yBsaW51eCB4ODZfNjQ7IHJ20jcyLjApIGdlY2tvLzIwMTAwMTAxIGZpcmVmb3gvNzIuMCAxMjcuMC4xLjE
gMTAuMDIuMjAyMCAwMTk6Mzk6MzQ="/>
                </property_list>
            </csd>
        </ss>
        <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="chunt" version="7"
description="Call hunt groups" active="false" enabled="true">
            <msd>

```

```

                                <property name="enumQueue_strategy"
type="string" description="Internal field to validate queue_strategy property" value="wait|
drop"/>
                                <property name="enumSearch_strategy"
type="string" description="Internal field to validate search_strategy property" value="first|
last|random|longest_idle"/>
                                <property name="regimeMode"
type="string" description="Internal field to validate search_strategy property"
value="group:numbers:queue_strategy|
serial:numbers:cyclic:search_strategy>window_shift_timeout:max_shift_count:queue_strategy|
longest_idle:numbers:cyclic>window_shift_timeout:max_shift_count:queue_strategy|
delay_group:numbers>window_shift_timeout:queue_strategy|manual"/>
                                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                    <item value="assistant"/>
                                    <item value="follow_me"/>
                                    <item value="cfu_type2"/>
                                    <item value="meet_me_conference"/>
                                    <item value="cfu"/>
                                    <item value="find_me"/>
                                    <item value="cf_aon"/>
                                    <item value="cft"/>
                                    <item value="cgg"/>
                                </property_list>
                                </msd>
                                <ssd>
                                <property name="forwarding_ss"
type="boolean" description="" value="true"/>
                                <property name="delay_retrieve_timeout
_3" type="integer" description="Delay retrieve timeout for waiting free group subscriber"
value="15000"/>
                                <property name="delay_retrieve_timeout
_2" type="integer" description="Delay retrieve timeout for release group call by slave
subscriber" value="5000"/>
                                <property name="delay_retrieve_timeout
_1" type="integer" description="Delay retrieve timeout for release group call by group
subscriber" value="1000"/>
                                </ssd>
                                <csd>
                                <property name="queue_size"
type="positive_integer" description="Wait calls' queue size" value="0"/>
                                <property name="search_strategy"
type="enum" description="(first | last | random | longest_idle)" value="first"/>
                                <property name="queue_strategy"
type="enum" description="(wait | drop)" value="wait"/>
                                <property name="window_shift"
type="positive_integer" description="The value on which shift window_start_pos when
window_shift_timeout fired" value="1"/>

```

```

                                <property name="window_start_pos"
type="integer" description="The index (zero-based) of the first subscriber in the window"
value="0"/>
                                <property name="window_size"
type="positive_integer" description="The number of simultaneous calls to the call-hunt's
subscribers" value="1"/>
                                <property name="max_shift_count"
type="integer" description="The number of times that a window can be shift" value="-1"/>
                                <property name="window_shift_timeout"
type="positive_integer" description="Interval after which window shift on the window_shift
value" value="0"/>
                                <property name="reset_window_start_pos
_after_call" type="boolean" description="Reset window start position after all incoming calls
will be connected with group calls" value="false"/>
                                <property name="cyclic" type="boolean"
description="Flag that is show is call hunt group cyclic or not" value="false"/>
                                <property name="mode" type="regime"
description="The type of the chunt"/>
                                <property_list name="numbers"
type="address" maxlen="10" description="Call-hunt member list">
                                    <undefined/>
                                </property_list>
                                </csd>
                                </ss>
                                <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="chold" version="8"
description="Call hold" active="false" enabled="true">
                                    <ssd>
                                <property name="dtmf_detector"
type="boolean" description="" value="true"/>
                                    </ssd>
                                </csd>
                                <property name="disable_moh"
type="boolean" description="Disable music on hold" value="false"/>
                                <property name="dtmf_sequence_as_flash
" type="boolean" description="Try interpret the DTMF sequence as a flash (look at incall
feature codes)" value="false"/>
                                </csd>
                                </ss>
                                <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cgg" version="7"
description="Group-calling" active="false" enabled="true">
                                    <msd>
                                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                    <item value="assistant"/>
                                    <item value="follow_me"/>
                                    <item value="cfu_type2"/>

```

```

        <item value="meet_me_conference"/>
        <item value="cfu"/>
        <item value="find_me"/>
        <item value="cf_aon"/>
        <item value="cft"/>
    </property_list>
</msd>
<ssd>
    <property name="forwarding_ss"
type="boolean" description="" value="true"/>
    </ssd>
<csd>
    <property name="send_call_diversion"
type="boolean" description="Send call diversion flag in call signalisation" value="true"/>
    <property_list name="numbers"
type="address" maxlen="10" description="Group's numbers">
        <undefined/>
    </property_list>
</csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfu_type2" version="14"
description="Call Forwarding Unconditional without notification" active="false"
enabled="true">
    <msd>
        <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
        <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss"
type="boolean" description="" value="true"/>
    </ssd>
<csd>
    <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
    <property name="external_number"
type="address" description="External forward number" value=""/>
    <property name="use_redirection_counte
r" type="boolean" description="Use redirectionCounter field to count hops" value="false"/>
    <property name="number" type="address"
description="Forward number"/>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"

```

```

xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfu" version="14"
description="Call Forwarding Unconditional" active="false" enabled="true">
  <msd>
    <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
    <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
      <item value="assistant"/>
      <item value="follow_me"/>
      <item value="cfu_type2"/>
      <item value="meet_me_conference"/>
    </property_list>
  </msd>
  <ssd>
    <property name="forwarding_ss"
type="boolean" description="" value="true"/>
  </ssd>
  <csd>
    <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
    <property name="external_number"
type="address" description="External forward number" value=""/>
    <property name="number" type="address"
description="Forward number"/>
  </csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cft" version="14"
description="Call Forwarding Unconditional Time Dependent" active="false" enabled="true">
  <msd>
    <property name="enumTime_format"
type="string" description="Internal field to validate time_format property" value="local|utc"/
>
    <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
    <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
      <item value="assistant"/>
      <item value="follow_me"/>
      <item value="cfu_type2"/>
      <item value="meet_me_conference"/>
      <item value="cfu"/>
      <item value="find_me"/>
      <item value="cf_aon"/>
    </property_list>
  </msd>
  <ssd>

```

```

type="boolean" description="" value="true"/>
                                <property name="forwarding_ss"
                                </ssd>
                                <csd>
                                    <property name="time_format"
type="enum" description="CFT for Sunday (local | utc)" value="local"/>
                                    <property name="day_7" type="term"
description="{{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                    <property name="day_6" type="term"
description="{{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                    <property name="day_5" type="term"
description="{{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                    <property name="day_4" type="term"
description="{{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                    <property name="day_3" type="term"
description="{{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                    <property name="day_2" type="term"
description="{{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                    <property name="day_1" type="term"
description="{{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                </csd>
                                </ss>
                                <ss domain="biysk.local" address="240101"
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfos" version="12"
description="Call Forwarding Out of Service" active="false" enabled="true">
                                <msd>
                                    <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
                                    <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                        <item value="cfu"/>
                                        <item value="assistant"/>
                                        <item value="follow_me"/>
                                        <item value="tsmn_request"/>
                                    </property_list>
                                </msd>
                                <ssd>
                                    <property name="forwarding_ss"
type="boolean" description="" value="true"/>
                                </ssd>
                                </csd>

```

```

        <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
        <property name="external_number"
type="address" description="External forward number" value=""/>
        <property name="number" type="address"
description="Forward number"/>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfnr_type2" version="11"
description="Call Forwarding No Reply without notification" active="false" enabled="true">
    <msd>
        <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
        <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="cfnr"/>
            <item value="follow_me"/>
            <item value="tsmn_request"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss"
type="boolean" description="" value="true"/>
    </ssd>
</csd>
<csd>
    <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
    <property name="external_number"
type="address" description="External forward number" value=""/>
    <property name="use_redirection_counte
r" type="boolean" description="Use redirectionCounter field to count hops" value="false"/>
    <property name="timeout"
type="positive_integer" description="Timeout before call forward (sec)" value="15"/>
    <property name="number" type="address"
description="Forward number"/>
</csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfnr" version="11"
description="Call Forwarding No Reply" active="false" enabled="true">
    <msd>
        <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
        <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">

```



```

        <item value="cfu"/>
        <item value="assistant"/>
        <item value="follow_me"/>
        <item value="tsmn_request"/>
    </property_list>
</msd>
<ssd>
    <property name="forwarding_ss"
type="boolean" description="" value="true"/>
</ssd>
<csd>
    <property name="timeout"
type="positive_integer" description="Timeout before call forward (sec)" value="15"/>
    <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
    <property name="external_number"
type="address" description="External forward number" value=""/>
    <property name="number" type="address"
description="Forward number"/>
</csd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfb" version="11"
description="Call Forwarding Busy" active="false" enabled="true">
    <msd>
        <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
        <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="tsmn_request"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss"
type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
        <property name="external_number"
type="address" description="External forward number" value=""/>
        <property name="number" type="address"
description="Forward number"/>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"

```

```

xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cf_sip" version="9"
description="Call Forwarding by SIP redirection (302 message)" active="true" enabled="true">
  <msd>
    <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
    <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
      <item value="cfu"/>
      <item value="assistant"/>
      <item value="follow_me"/>
      <item value="tsmn_request"/>
    </property_list>
  </msd>
  <ssd>
    <property name="forwarding_ss"
type="boolean" description="" value="true"/>
  </ssd>
</ss>
<ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cf_aon" version="9"
description="Call Forwarding AON" active="false" enabled="true">
  <msd>
    <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
    <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
      <item value="assistant"/>
      <item value="follow_me"/>
      <item value="cfu_type2"/>
      <item value="meet_me_conference"/>
      <item value="cfu"/>
      <item value="find_me"/>
    </property_list>
  </msd>
  <ssd>
    <property name="forwarding_ss"
type="boolean" description="" value="true"/>
  </ssd>
  <csd>
    <property name="number" type="address"
description="Forward number"/>
    <property name="timeout"
type="positive_integer" description="Timeout before call forward (sec)" value="15"/>
    <property name="no_reply"
type="boolean" description="Activate service when subscriber no reply" value="false"/>
    <property_list name="prefix_number"
type="address" maxlen="10" description="Prefix number A">
      <undefined/>

```

```

        </property_list>
    </csd>
</ss>
<ss domain="biysk.local" address="240101"
    xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cc_agent" version="2"
    description="Contact-Center agent" active="true" enabled="true"/>
    <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="callback" version="5"
        description="CallBack" active="false" enabled="true">
        <ssd>
            <property name="dtmf_detector"
                type="boolean" description="" value="true"/>
            <property name="max_call_retry"
                type="positive_integer" description="The maximum count of call's retry to initial subscriber"
                value="5"/>
            <property name="a_onhook_timeout"
                type="positive_integer" description="Timeout for A subscriber onhook (sec)" value="5"/>
            <property name="collect_digits_timeout" type="integer" description="Wait timeout to collect cancel digits" value="60"/>
        </ssd>
    </ss>
    <ss domain="biysk.local" address="240101"
        xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="call_recording" version="11"
        description="Call recording" active="false" enabled="true">
        <msd>
            <property name="enumMode"
                type="string" description="Internal field to validate mode property" value="always_on|
on_demand|after_answer"/>
            <property name="owner" type="atom"
                description="SS type. Available values: user | trunk | any" value="any"/>
        </msd>
        <ssd>
            <property name="dtmf_detector"
                type="boolean" description="" value="true"/>
            <property name="locked"
                type="boolean" description="Can subscriber manage current SS, or not" value="true"/>
        </ssd>
    </csd>
    <property name="separated"
        type="boolean" description="The separated recording of the call. If true, the outgoing audio
of the subscriber is recorded to first audio track, the incoming audio of the subscriber is
recorded to second audio track. If false, the incoming and the outgoing audio of the
subscriber is recorded to first audio track." value="false"/>
    <property name="mode" type="enum"
        description="The mode of the recording call (always_on|on_demand)"/>
    </csd>
</ss>

```

```

<ss domain="biysk.local" address="240101"
  xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="boss_group" version="1"
  description="Boss-group" active="false" enabled="true">
  <msd>
    <property name="integer_rangelines
_count" type="string" description="Internal field" value="[0,16]"/>
    <property name="integer_rangeDelay
_ring_timeout" type="string" description="Internal field" value="[0,60]"/>
    <property_list name="dependency_li
st" type="atom" maxlen="0" description="List of SS that is depended from SS">
      <item value="auto_redial"/>
    </property_list>
    <property_list name="conflict_list
" type="atom" maxlen="0" description="List of SS that is conflict with current SS">
      <item value="cw"/>
    </property_list>
  </msd>
  <ssd>
    <property name="dtmf_detector"
type="boolean" description="" value="true"/>
  </ssd>
  <csd>
    <property name="call_waiting_tone"
type="boolean" description="Play call waiting tone for second(s) calls on group"
value="false"/>
    <property name="members_fwd_busy_o
verride" type="boolean" description="Ignore members CFB for group calls" value="true"/>
    <property name="lines_count"
type="integer_range" description="Incoming boss-group lines count" value="2"/>
    <property name="delay_ring_timeout"
type="integer_range" description="Timeout before call on delayed members" value="0"/>
    <property name="display_name"
type="string" description="Display name for boss group" value=""/>
    <property_list name="silent_member
s" type="address" maxlen="8" description="Members for BLF and pickup only group calls">
      <undefined/>
    </property_list>
    <property_list name="delay_members
" type="address" maxlen="8" description="Members for delay call">
      <undefined/>
    </property_list>
    <property_list name="members"
type="address" maxlen="8" description="Members for immediate call">
      <undefined/>
    </property_list>
  </csd>
</ss>
<ss domain="biysk.local" address="240101"

```

```

                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="blf" version="3"
description="Busy Lamp Field" active="false" enabled="true">
                                <csd>
                                    <property name="ignore_page_calls"
type="boolean" description="Doesn't send BLF notification from Zone Page calls" value="true"/>
                                >
                                    <property name="max_subscribe"
type="positive_integer" description="Maximum number of subscriptions" value="16"/>
                                </csd>
                                </ss>
                                <ss domain="biysk.local" address="240101"
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="avatar" version="2"
description="Picture for outgoing calls" active="false" enabled="true">
                                <csd>
                                    <property name="pic_addr"
type="uri_string" description="URL to access the image" value=""/>
                                </csd>
                                </ss>
                                <ss domain="biysk.local" address="240101"
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="auto_redial_with_callback"
version="5" description="Auto redial with callback" active="false" enabled="true">
                                <msd>
                                    <property name="enumRingback"
type="string" description="Ringback" value="tone|moh"/>
                                </msd>
                                <csd>
                                    <property name="ringback"
type="enum" description="Ringback (tone | moh)" value="tone"/>
                                    <property name="b_no_answer_timeou
t" type="positive_integer" description="Awaiting A-side answer timeout (sec)" value="30"/>
                                    <property name="a_no_answer_timeou
t" type="positive_integer" description="Awaiting B-side answer timeout (sec)" value="10"/>
                                    <property name="recall_count"
type="positive_integer" description="Recalls' count" value="30"/>
                                    <property name="recall_timer"
type="positive_integer" description="Time between recalls (sec)" value="90"/>
                                </csd>
                                </ss>
                                <ss domain="biysk.local" address="240101"
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="auto_redial" version="2"
description="Auto redial" active="false" enabled="true">
                                <csd>
                                    <property name="no_answer_timeout"
type="positive_integer" description="Awaiting answer timeout (sec)" value="30"/>
                                    <property name="recall_count"
type="positive_integer" description="Recalls' count" value="6"/>

```

```

                                <property name="recall_timeout"
type="positive_integer" description="Timeout between recalls (sec)" value="15"/>
                                </csd>
                                </ss>
                                <ss domain="biysk.local" address="240101"
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="assistant" version="2"
                                description="Chief-assistant group" active="false" enabled="true">
                                <msd>
                                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="user"/>
                                </msd>
                                <ssd>
                                <property name="forwarding_ss"
type="boolean" description="" value="true"/>
                                </ssd>
                                <csd>
                                <property_list name="whitelist"
type="index_address" maxlen="128" description="White list members. Format: <address>[/
<index>]">
                                <undefined/>
                                </property_list>
                                <property_list name="assistants"
type="index_address" maxlen="4" description="Assistants numbers. Format: <address>[/
<index>]">
                                <undefined/>
                                </property_list>
                                </csd>
                                </ss>
                                <ss domain="biysk.local" address="240101"
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="alarm" version="8"
                                description="Alarm Call" active="false" enabled="true">
                                <msd>
                                <property name="handle_module"
type="atom" description="" value="ss_alarm_handler"/>
                                </msd>
                                <ssd>
                                <property name="collect_digits_tim
eout" type="integer" description="Wait timeout to collect cancel digits" value="60"/>
                                </ssd>
                                <csd>
                                <property name="alarm_9"
type="term" description="" value="off"/>
                                <property name="alarm_8"
type="term" description="" value="off"/>

```

```

                                <property name="alarm_7"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="alarm_6"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="alarm_5"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="alarm_4"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="alarm_3"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="alarm_2"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="alarm_1"
type="term" description=""{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "]" , "{"HH, MM"}" [,
AlarmName]}"" value="off"/>
                                <property name="cancel_digits"
type="address" description="Digits to cancel alarm" value="*"/>
                                <property name="call_attempt_timeo
ut" type="integer" description="Awaiting timer between calls' attempts" value="60"/>
                                <property name="call_attempt_count
" type="integer" description="The count of the call's attempt" value="3"/>
                                <property name="no_answer_timeout"
type="integer" description="Awaiting answer timer (value in seconds)" value="45"/>
                                </csd>
                                </ss>
                                <ss domain="biysk.local" address="240101"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="acb" version="2"
description="Anonymous Call Blocking" active="false" enabled="true"/>
                                <ss domain="biysk.local"
address="240101"
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="3way" version="6"
description="3-Way Conference" active="false" enabled="true">
                                <msd>
                                <property_list name="conflict_
list" type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                <item value="conference"/>
                                </property_list>
                                <property_list name="dependenc
y_list" type="atom" maxlen="0" description="List of SS that is depended from SS">
                                <item value="chold"/>
                                </property_list>
                                </msd>

```

```

type="boolean" description="" value="true"/>
        <ssd>
            <property name="dtmf_detector"
        </ssd>
    </ss>
</sss>
</out>

```

Пример 2:

Запросить информацию по определенному ДВО:

Запрос: http://192.168.1.12:9999/commands/ss_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_info.xsd">
  <request domain="biysk.local" address="240464" ss_name="chold"/>
</in>

```

Ответ:200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_info.xsd">
  <sss>
    <ss domain="biysk.local" address="240464"
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="chold" version="8" description="Call hold"
      active="true" enabled="true">
      <ssd>
        <property name="dtmf_detector" type="boolean" description="" value="true"/>
      </ssd>
      <csd>
        <property name="dtmf_sequence_as_flash" type="boolean" description="Try interpret
the DTMF sequence as a flash (look at incall feature codes)" value="false"/>
        <property name="disable_moh" type="boolean" description="Disable music on hold"
value="false"/>
      </csd>
    </ss>
  </sss>
</out>

```


Нс_ss_install

Команда предназначена для инсталляции услуг из XML-файлов, описывающих работу услуг.

Аналог команды в CoCon:

`/cluster/storage/<CLUSTER>/ss/install`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/ss_install`

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            <xs:attribute name="file" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

Пример:

Запрос:

http://192.168.1.21:9999/commands/ss_install

Параметры:

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ss_install.xsd">
  <request file="ss_blf.xml" storage="ds1"/>
</in>
```

Ответ-204:Пустой xml.

Нс_ss_show

Команда позволяет просмотреть список установленных в систему услуг.

Аналог команды в CoCon:

cluster/**s**torage/<CLUSTER>/**s**s/**s**how

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_show

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с запросом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="ss.xsd"/>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            </xs:complexType>
          </xs:element>
        </xs:all>
      </xs:complexType>
    </xs:element>
  <!-- out-->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="sss" type="ssListType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/ss_show

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_show.xsd">
  <request storage="ds1"/>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss_show.xsd">
  <sss>
    <ss
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="zone_page" version="10" description="Zone Page
      call" active="false" enabled="false">
      <msd>
        <property name="enumPriority" type="string" description="Call's priority"
        value="0_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
        <property name="enumInitiation_policy" type="string" description="Initiation
        call's policy" value="managers|members|any"/>
        <property name="integer_rangeMulticast_port" type="string"
        description="Internal field" value="[0,65535]"/>
        <property name="integer_rangeMulticast_ttl" type="string"
        description="Internal field" value="[1,255]"/>
        <property name="enumMulticast_codec" type="string" description="Internal
        field" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|G726-40"/>
      </msd>
      <ssd>
        <property name="dtmf_detector" type="boolean" description="" value="true"/>
      </ssd>
      <csd>
        <property name="priority" type="enum" description="(0_flash_override | 1_flash
        | 2_immediate | 3_priority | 4_routine)" value="3_priority"/>
        <property name="intro_pause" type="integer" description="Pause before play
        intro files" value="3"/>
        <property name="announcement_pause" type="integer" description="Pause before
        play announcement files" value="0"/>
        <property name="announcement_playback_cnt" type="integer"
        description="Announcement files playback count" value="1"/>
        <property name="announcement_repeat_time" type="integer"
        description="Announcement files repeat timeout" value="10"/>
        <property name="terminate_when_initiator_disconnects" type="boolean"
        description="Terminate call after initiator disconnected"/>
        <property name="terminate_when_playback_ends" type="boolean"
        description="Terminate call after playback ends"/>
        <property name="terminate_when_last_listener_leave" type="boolean"
        description="Terminate call after last listener disconnects"/>
        <property name="initiation_policy" type="enum" description="Initiation call's
        policy" value="any"/>
        <property name="display_name" type="string" description="ZonePage call
        DisplayName" value="ZonePage"/>
        <property name="play_release_tone" type="boolean" description="ZonePage
        release tone" value="false"/>
        <property name="announcement_repeat_digit" type="address" description="Digit
        to repeat announcement files" value="#"/>

```

```

        <property name="multicast_ip" type="ip" description="Optional multicast
address to use for announcement" value="0.0.0.0"/>
        <property name="multicast_port" type="integer_range" description="Port of
multicast address" value="0"/>
        <property name="multicast_codec" type="enum" description="Codec to be used for
multicasting" value="PCMA"/>
        <property name="multicast_ttl" type="integer_range" description="TTL for
multicast" value="1"/>
        <property_list name="members" type="address" maxlen="256" description="Zone
page member list">
            <undefined/>
        </property_list>
        <property_list name="managers" type="address" maxlen="256" description="Zone
page managers list"/>
        <property_list name="intro_files" type="media_resource" maxlen="10"
description="List of Intro media_files"/>
        <property_list name="announcement_files" type="media_resource" maxlen="10"
description="List of Announcement media_files"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="walkie_talkie" version="1" description="Walkie
Talkie" active="false" enabled="false">
    <ssd>
        <property name="dtmf_detector" type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="swap_digit" type="address" description="Digit to swap
reception / transmission mode" value="*"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="voicemail" version="13" description="Voice mail
service" active="false" enabled="false">
    <msd>
        <property_list name="conflict_list" type="atom" maxlen="0" description="List
of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
            <item value="cfu"/>
            <item value="find_me"/>
            <item value="cf_aon"/>
            <item value="cft"/>
        </property_list>
    </msd>
</ssd>

```

```

        <property name="dtmf_detector" type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="no_reply" type="boolean" description="Activate service when
subscriber no reply" value="true"/>
        <property name="busy" type="boolean" description="Activate service when
subscriber is busy" value="false"/>
        <property name="dnd" type="boolean" description="Activate service when
subscriber activate DND" value="false"/>
        <property name="out_of_service" type="boolean" description="Activate service
when subscriber is out of service" value="true"/>
        <property name="unconditional" type="boolean" description="Activate service
any time" value="false"/>
        <property name="timeout" type="positive_integer" description="Timeout before
call will be forwarded at voicemail (sec)" value="30"/>
        <property name="send_by_email" type="boolean" description="Send recorded
message by email" value="false"/>
        <property name="min_message_duration" type="positive_integer" description="The
minimum voice mail duration (sec). In case of voice mail duration will be less that
min_message_duration, this message will be dropped" value="0"/>
        <property name="max_message_duration" type="positive_integer" description="The
maximum voice mail duration (sec). 0 - in case of not limited duration" value="300"/>
        <property name="mailbox_volume" type="positive_integer" description="The
mailbox volume (sec)" value="1800"/>
        <property name="override_messages" type="boolean" description="Override
messages in case of mailbox is full" value="true"/>
        <property name="email_from" type="string" description="Email from template.
Macro variables: %DATE%, %TIME%, %CALLING%, %DISPLAYNAME%" value="%DISPLAYNAME%"/>
        <property name="email_subject" type="string" description="Email subject
template. Macro variables: %DATE%, %TIME%, %CALLING%, %DISPLAYNAME%" value="Voicemail from
%CALLING% at %DATE% %TIME%"/>
        <property name="email_body" type="string" description="Email body template.
Macro variables: %DATE%, %TIME%, %CALLING%, %DISPLAYNAME%" value=""/>
        <property name="play_message_details" type="boolean" description="Play
message's details (number, time) before message" value="false"/>
        <property name="password" type="address" description="There is password to
have access to the voicemail box" value=""/>
        <property name="max_silence" type="positive_integer" description="There is
time(sec) while service wait for a contiguous period of silence before terminating an incoming
call to voice mail" value="0"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="voice_page_control" version="2"
    description="'Voice Page Control' service" active="false" enabled="false">
    <csd>
        <property name="accept_incoming_vp" type="boolean" description="Accept or
Reject voice page calls" value="true"/>
    </csd>
</ss>

```

```

        <property name="block_incoming_vp_by_dnd" type="boolean" description="Block
voice page calls by DND service" value="true"/>
        <property name="mute_incoming_vp" type="boolean" description="Accept incoming
voice page call in silent mode" value="true"/>
    </csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="voice_page" version="9" description="Voice Page
call" active="false" enabled="false">
  <msd>
    <property name="enumPriority" type="string" description="Call's priority"
value="@_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
  </msd>
  <ssd>
    <property name="dtmf_detector" type="boolean" description="" value="true"/>
  </ssd>
  <csd>
    <property name="skip_cf" type="boolean" description="Skipping call forwarding"
value="true"/>
    <property name="priority" type="enum" description="(@_flash_override | 1_flash
| 2_immediate | 3_priority | 4_routine)" value="3_priority"/>
    <property name="announcement_repeat_time" type="integer"
description="Announcement files repeat timeout" value="10"/>
    <property name="announcement_playback_cnt" type="integer"
description="Announcement files playback count [0..5]" value="0"/>
    <property name="swap_digit" type="address" description="Digit to swap
reception / transmission mode" value="*/>
    <property name="announcement_repeat_digit" type="address" description="Digit
to repeat announcement files" value="#"/>
    <property_list name="intro_files" type="media_resource" maxlen="10"
description="List of Intro media_files"/>
    <property_list name="announcement_files" type="media_resource" maxlen="10"
description="List of Announcement media_files"/>
  </csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="vip_call" version="4" description="Vip call"
  active="false" enabled="false">
  <msd>
    <property name="enumPriority" type="string" description="Call's priority"
value="@_flash_override|1_flash|2_immediate|3_priority|4_routine"/>
  </msd>
  <csd>
    <property name="priority" type="enum" description="(@_flash_override | 1_flash
| 2_immediate | 3_priority | 4_routine)" value="4_routine"/>
    <property name="implicit_mode" type="boolean" description="Use vip call
without feature code." value="false"/>

```

```

        <property name="skip_low_priority_call" type="boolean" description="Skipping
second call if he has with low priority." value="false"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="tsmn_request" version="3" description="Support
request on TSMN" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available values:
user | trunk | any" value="trunk"/>
    </msd>
    <csd>
        <property name="timeout" type="positive_integer" description="Timeout on TSMN
request(s) (msec)" value="3000"/>
        <property name="tsmn_trunk" type="trunk" description="Trunk to TSMN server"/>
        <property name="tsmn_trunk_backup" type="trunk" description="Backup trunk to
TSMN server" value=""/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="teleconference_manager" version="3"
description="Teleconference manager" active="false" enabled="false">
    <csd>
        <property_list name="second_line" type="address" maxlen="10"
description="Second line numbers"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="teleconference" version="4"
description="Teleconference" active="false" enabled="false">
    <msd>
        <property name="enumType" type="string" description="Teleconference type:
meeting or room" value="meeting|room"/>
    </msd>
    <csd>
        <property name="destroy_by_no_initiator" type="boolean" description="Destroy
the teleconference by release from the initiator?" value="true"/>
        <property name="template_from_room" type="teleconference_room"
description="Get template from the teleconference room" value=""/>
        <property name="type" type="enum" description="Teleconference type: meeting or
room" value="meeting"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="speed_dial" version="5" description="Speed dial
(Abbreviated dialing)" active="false" enabled="false">

```



```

    <msd>
      <property name="enumType" type="string" description="The type of address book.
address_book_10 supports 10 number in the address book. address_book_100 supports 100 numbers
on the address book" value="address_book_10|address_book_100"/>
    </msd>
    <csd>
      <property name="type" type="enum" description="(address_book_10 |
address_book_100)" value="address_book_100"/>
      <property_list name="map" type="speed_dial_item" maxlen="50" description="Map
between short dial and full address. Format: <short_dial>/<address>"/>
    </csd>
  </ss>
  <ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="smart_cancel" version="2" description="Smart
cancel service" active="false" enabled="false">
    <msd>
      <property name="integer_rangeTimeout" type="string" description="Timeout range
property" value="[0,5]"/>
    </msd>
    <csd>
      <property name="timeout" type="integer_range" description="Timeout to wait
release" value="2"/>
    </csd>
  </ss>
  <ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="sip_message" version="1" description="SIP Message
service" active="false" enabled="false">
    <msd>
      <property name="enumMode" type="string" description="Call's priority"
value="sendonly|recvonly|sendrecv"/>
    </msd>
    <csd>
      <property name="mode" type="enum" description="(sendonly | recvonly |
sendrecv)" value="sendrecv"/>
    </csd>
  </ss>
  <ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="second_handset" version="1" description="Second
Handset" active="false" enabled="false"/>
    <ss
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="scr" version="2" description="Selective Call
Rejection" active="false" enabled="false">
      <csd>
        <property_list name="blacklist" type="index_address" maxlen="50"
description="Black list members. Format: <address>[/<index>]"/>
      </csd>
    </ss>
  </ss>
  <undefined/>

```

```

        </property_list>
    </csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="sco_white" version="2" description="Selective
Call Originating, White list" active="false" enabled="false">
  <csd>
    <property_list name="whitelist" type="index_address" maxlen="50"
description="White list members. Format: <address>[/<index>]">
      <undefined/>
    </property_list>
  </csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="sco_black" version="2" description="Selective
Call Originating, Black list" active="false" enabled="false">
  <csd>
    <property_list name="blacklist" type="index_address" maxlen="50"
description="Black list members. Format: <address>[/<index>]">
      <undefined/>
    </property_list>
  </csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="sca" version="2" description="Selective Call
Acceptance" active="false" enabled="false">
  <csd>
    <property_list name="whitelist" type="index_address" maxlen="50"
description="White list members. Format: <address>[/<index>]">
      <undefined/>
    </property_list>
  </csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="ring_back_tone" version="2" description="Ring
back tone" active="false" enabled="false">
  <csd>
    <property name="file" type="media_resource" description="File to play on
ring"/>
  </csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ss.xsd" name="rfc" version="2" description="Rejection of
Forwarded Calls" active="false" enabled="false"/>
</ss>

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="remote_ring" version="1" description="Remote ring
to callee" active="false" enabled="false">
        <msd>
            <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
                <item value="distinctive_ring"/>
            </property_list>
        </msd>
        <csd>
            <property_list name="announcement_files" type="media_resource"
maxlen="10" description="List of Announcement media_files"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="remote_phone" version="1" description="Remote
phone" active="false" enabled="false">
        <msd>
            <property name="handle_module" type="atom" description=""
value="ss_remote_phone_handler"/>
        </msd>
        <csd>
            <property name="remote_endpoint" type="interface" description="Remote
interface" value=""/>
            <property name="pin" type="address" description="PIN-code"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="redial" version="2" description="Last number
redial" active="false" enabled="false"/>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="rbp" version="3" description="SS for restricting
call-processes by NI and PIN" active="false" enabled="false">
        <csd>
            <property name="implicit_call" type="boolean" description="Use
implicit RBP call (Request PIN-code in case of use recsricted call)" value="false"/>
            <property_list name="restricted_ni" type="ni" maxlen="6"
description="Restricted number indication">
                <undefined/>
            </property_list>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfu" version="3" description="QSIG: Call
Forwarding Unconditional" active="false" enabled="false">
        <msd>

```

```

        <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="trunk"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="follow_me"/>
        </property_list>
    </msd>
    <csd>
        <property name="hide_forwarding" type="boolean" description="Hide
forwarding fields (rgn, ocdrn)" value="false"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfnr" version="2" description="QSIG: Call
Forwarding No Reply" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available
values:user | trunk | any" value="trunk"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="cfu"/>
        </property_list>
    </msd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="qsig_cfb" version="3" description="QSIG: Call
Forwarding Busy" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="trunk"/>
        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
            <item value="cfu"/>
        </property_list>
    </msd>
    <csd>
        <property name="hide_forwarding" type="boolean" description="Hide
forwarding fields (rgn, ocdrn)" value="false"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="ss.xsd" name="privacy" version="2" description="Privacy"
    active="false" enabled="false">
    <msd>
        <property name="enumMode" type="string" description="Mode"
value="on_demand|always_on"/>

```

```

        </msd>
        <ssd>
            <property name="dtmf_detector" type="boolean" description=""
value="true"/>
        </ssd>
        <csd>
            <property name="mode" type="enum" description="Privacy usage mode"
value="on_demand"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="presence" version="2" description="SIP Presence"
active="false" enabled="false">
        <csd>
            <property name="max_subscribe" type="positive_integer"
description="Maximum number of subscriptions" value="16"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="pickup" version="2" description="Call Pickup"
active="false" enabled="false">
        <csd>
            <property_list name="pickup_groups" type="index_string"
maxlen="10" description="Call pick-up's groups names. Format: <Name>[/<Inx>]"/>
                <undefined/>
            </property_list>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="personal_ivr" version="2" description="Personal
IVR script" active="false" enabled="false">
        <msd>
            <property name="handle_module" type="atom" description=""
value="ss_personal_ivr_handler"/>
            <property name="owner" type="atom" description="SS type. Available
values: user | trunk | any" value="user"/>
        </msd>
        <csd>
            <property name="script" type="ivr_script_id" description="Script
ID" value=""/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="park" version="2" description="Call park"
active="false" enabled="false"/>
    </ss>

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="multicast_page" version="1"
        description="Multicast IP Paging" active="false" enabled="false">
        <msd>
            <property name="enumCodec" type="string" description="Internal
            field" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|G726-40"/>
            <property name="integer_rangePort" type="string"
            description="Internal field" value="[10000,65535]"/>
            <property name="integer_rangeTtl" type="string"
            description="Internal field" value="[1,255]"/>
        </msd>
        <csd>
            <property name="ip" type="ip" description="IP address to cast
            the media to"/>
            <property name="port" type="integer_range" description="UDP
            port to cast the media to (10000-65535)"/>
            <property name="codec" type="enum" description="Codec to be
            used for multicasting" value="PCMA"/>
            <property name="ttl" type="integer_range" description="TTL for
            multicast packets (1-255)" value="1"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="multicast_listen" version="1"
        description="Multicast Listen Service" active="false" enabled="false">
        <msd>
            <property name="enumCodec" type="string" description="Internal
            field to codec property" value="G722|PCMU|PCMA|G729|G723_53|G723_63|G726-16|G726-24|G726-32|
            G726-40"/>
            <property name="integer_rangePort" type="string"
            description="Internal field to udp port property" value="[10000,65535]"/>
        </msd>
        <csd>
            <property name="ip" type="ip" description="Ip address for
            listen multicast."/>
            <property name="port" type="integer_range" description="Port
            for listen multicast (10000-65535)."/>
            <property name="codec" type="enum" description="Audio codec
            for multicast traffic." value="G722"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="m1" version="4" description="Multiline"
        active="false" enabled="false">
        <msd>
            <property name="integer_rangeLine_count" type="string"
            description="Internal field" value="[0,16]"/>

```

```

                <property name="integer_rangeLine_count_out" type="string"
description="Internal field" value="[-1,16]"/>
            </msd>
            <csd>
                <property name="line_count" type="integer_range"
description="Line count" value="3"/>
                <property name="line_count_out" type="integer_range"
description="Outgoing lines count" value="-1"/>
            </csd>
        </ss>
        <ss>
            xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
            xs:noNamespaceSchemaLocation="ss.xsd" name="message" version="2" description="Message
service" active="false" enabled="false">
                <msd>
                    <property name="owner" type="atom" description="SS type.
Available values: user | trunk | any" value="any"/>
                </msd>
                <ssd>
                    <property name="message_box_size" type="integer"
description="Message box size" value="20"/>
                    <property name="remove_only_after_answer" type="boolean"
description="Remove message only after abonent answer" value="false"/>
                </ssd>
            </ss>
            <ss>
                xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
                xs:noNamespaceSchemaLocation="ss.xsd" name="meet_me_conference" version="2" description="Meet
Me Conference" active="false" enabled="false">
                    <msd>
                        <property name="regimeMode" type="string"
description="Internal field to validate mode property" value="master:destroy_mode:room_number|
member:room_number"/>
                        <property name="enumDestroy_mode" type="string"
description="Internal field to validate mode property" value="by_no_master|by_no_more_calls"/>
                        <property_list name="conflict_list" type="atom" maxlen="0"
description="List of SS that is conflict with current SS">
                            <item value="assistant"/>
                            <item value="follow_me"/>
                            <item value="cfu_type2"/>
                        </property_list>
                    </msd>
                    <csd>
                        <property name="mode" type="regime" description="The mode of
meet me conference (master|member)"/>
                        <property name="destroy_mode" type="enum" description="The
mode of destroying conference (by_no_master|by_no_more_calls)" value="by_no_more_calls"/>
                        <property name="room_number" type="address" description="Room
number" value=""/>
                    </csd>
                </ss>
            </ss>
        </ss>
    </ss>

```

```

    </ss>
    <ss
      xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
      xs:noNamespaceSchemaLocation="ss.xsd" name="mcid" version="2" description="Malicious Call
      Identification" active="false" enabled="false"/>
      <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="introduce" version="1" description="Introduce
        announcement from callee" active="false" enabled="false">
          <csd>
            <property name="announcement" type="media_resource"
            description="Announcement media file"/>
          </csd>
        </ss>
      <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="intervention" version="4"
        description="Intervention" active="false" enabled="false">
          <msd>
            <property name="enumRestriction" type="string"
            description="Internal field to validate mode property" value="full_control|
            implicitly_allowed|explicitly_allowed"/>
          </msd>
          <ssd>
            <property name="dtmf_detector" type="boolean"
            description="" value="true"/>
          </ssd>
          <csd>
            <property name="restriction" type="enum" description="The
            restriction to the intervention call (full_control|allowed_only)" value="full_control"/>
          </csd>
        </ss>
      <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="intercom" version="1" description="Intercom"
        active="false" enabled="false"/>
      <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="hide_cf_name" version="1" description="Hide call
        forwarding name for subscriber A" active="false" enabled="false"/>
      <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="fxo_trunk" version="1" description="FX0 trunk
        service" active="false" enabled="false">
          <msd>
            <property name="owner" type="atom" description="SS
            type. Available values: user | trunk | any" value="trunk"/>
          </msd>
        </ss>
      <ss

```



```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="follow_me_no_response" version="8"
description="Follow me no response" active="false" enabled="false">
    <msd>
        <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="cft"/>
            <item value="cgg"/>
            <item value="follow_me"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address"
description="Followed number" value=""/>
        <property name="timeout" type="positive_integer"
description="Timeout before call forward (sec)" value="20"/>
        <property name="pin" type="address"
description="PIN-code" value=""/>
        <property name="external_number" type="address"
description="External forward number" value=""/>
        <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
    </csd>
</ss>
<ss>
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="follow_me" version="11" description="Follow me"
active="false" enabled="false">
    <msd>
        <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
            <item value="assistant"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="number" type="address"
description="Followed number" value=""/>
        <property name="pin" type="address"
description="PIN-code" value=""/>

```

```

        <property name="external_number" type="address"
description="External forward number" value=""/>
        <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="flexicall" version="1" description="FlexiCall"
active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS
type. Available values: user | trunk | any." value="user"/>
        <property name="enumMode" type="string"
description="Internal field to validate mode property" value="all|external|internal"/>
        <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
            <item value="cgg"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="mode" type="enum"
description="Mode operating. Available values (all | external | internal)"/>
        <property_list name="external_numbers"
type="address" maxlen="10" description="Group's external numbers"/>
        <property_list name="internal_numbers"
type="address" maxlen="10" description="Group's internal numbers"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="find_me_no_response" version="6"
description="Find Me no response" active="false" enabled="false">
    <msd>
        <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="ctu"/>
            <item value="cgg"/>
            <item value="follow_me"/>
            <item value="follow_me_no_response"/>
            <item value="find_me"/>
        </property_list>
    </msd>
    <ssd>

```

```

        <property name="forwarding_ss" type="boolean"
description="" value="false"/>
    </ssd>
    <csd>
        <property name="send_call_diversion"
type="boolean" description="Send call diversion flag in call signalisation" value="true"/>
        <property name="timeout" type="positive_integer"
description="Timeout before call forward (sec)" value="20"/>
        <property_list name="groups" type="find_me_group"
maxlen="32" description="Find me groups"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="find_me" version="7" description="Find Me"
active="false" enabled="false">
    <msd>
        <property_list name="conflict_list" type="atom"
maxlen="0" description="List of SS that is conflict with current SS">
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="cfu_type2"/>
            <item value="meet_me_conference"/>
            <item value="cfu"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss" type="boolean"
description="" value="true"/>
    </ssd>
    <csd>
        <property name="send_call_diversion"
type="boolean" description="Send call diversion flag in call signalisation" value="true"/>
        <property_list name="groups" type="find_me_group"
maxlen="32" description="Find me groups"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="fbc" version="1" description="Forwarding Barring
Call" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom" description="SS
type. Available values: user | trunk | any" value="any"/>
    </msd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="ss.xsd" name="fax_to_email" version="2" description="Fax to
email" active="false" enabled="false">

```

```

        <csd>
            <property name="email" type="string"
description="Email address" value="" />
            <property name="send_error_report" type="boolean"
description="Send email in case of fax received failed" value="false" />
        </csd>
    </ss>
    <ss>
        xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
        xs:noNamespaceSchemaLocation="ss.xsd" name="fax_receiver" version="2" description="Enable fax
receiving ability for terminals which is not support fax by their own" active="false"
enabled="false" />
        <ss>
            xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="dnd" version="4" description="Do not
Disturb" active="false" enabled="false">
                <msd>
                    <property name="enumMode" type="string"
description="Internal field to validate mode property" value="default|smart" />
                </msd>
                <csd>
                    <property name="mode" type="enum"
description="The mode of the working dnd service (default|smart)" value="default" />
                    <property_list name="whitelist"
type="index_address" maxlen="10" description="White list members. Format: <address>[/<index>]"
/>
                </csd>
            </ss>
            <ss>
                xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="distinctive_ring" version="4"
description="Destination ringtone change" active="false" enabled="false">
                    <csd>
                        <property_list name="filters"
type="distinctive_ring" maxlen="16" description="Distinctive ring for specific numbers" />
                    </csd>
                </ss>
                <ss>
                    xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="distinctive_picture" version="2"
description="Display caller image by groups" active="false" enabled="false">
                        <csd>
                            <property_list name="groups"
type="distinctive_picture" maxlen="20" description="Distinctive picture for specific numbers" /
>
                        </csd>
                    </ss>
                </ss>
            </ss>
        </ss>
    </ss>

```

```

        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="direct_call" version="2"
description="Direct call" active="false" enabled="false">
    <csd>
        <property name="number" type="address"
description="Call number to direct call"/>
        <property name="timeout"
type="positive_integer" description="Timeout before direct call (sec)" value="5"/>
    </csd>
</ss>
<ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cw" version="5" description="Call
Waiting" active="false" enabled="false">
    <msd>
        <property_list name="dependency_list"
type="atom" maxlen="0" description="List of SS that is depended from SS">
            <item value="chold"/>
        </property_list>
    </msd>
    <ssd>
        <property name="dtmf_detector" type="boolean"
description="" value="true"/>
    </ssd>
</ss>
<ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="ctr" version="5" description="Call
Transfer" active="false" enabled="false">
    <msd>
        <property_list name="dependency_list"
type="atom" maxlen="0" description="List of SS that is depended from SS">
            <item value="chold"/>
        </property_list>
    </msd>
    <ssd>
        <property name="dtmf_detector" type="boolean"
description="" value="true"/>
    </ssd>
</ss>
<ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cpip" version="2" description="Calling
Picture Identification Presentation" active="false" enabled="false">
    <msd>
        <property name="enumDisplay_mode"
type="string" description="Internal field to validate 'display-mode' property" value="icon|
fullscreen"/>
    </msd>
</csd>

```

```

        <property name="display_mode" type="enum"
description="Sets the picture display mode: icon mode or fullscreen mode." value="icon"/>
    </csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="conference" version="9"
description="Conference Call, Add-on" active="false" enabled="false">
    <msd>
        <property name="enumDestroy_mode"
type="string" description="Internal field to validate mode property" value="by_no_master|
by_no_more_calls"/>
        <property_list name="dependency_list"
type="atom" maxlen="0" description="List of SS that is depended from SS">
            <item value="chold"/>
            <item value="ctr"/>
        </property_list>
        <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
            <item value="3way"/>
        </property_list>
    </msd>
</csd>
    <property name="destroy_mode" type="enum"
description="The mode of destroying conference (by_no_master|by_no_more_calls)"
value="by_no_master"/>
    <property name="max_participants"
type="positive_integer" description="The maximum number of participants in the conference,
limited to the number 64 at the top. By default 16" value="16"/>
</csd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cnip" version="3" description="Calling
Name Identification Presentation" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
    </msd>
</ss>
<ss
    xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cliro" version="2" description="Calling
Line Identification Restriction Override" active="false" enabled="false"/>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="clir" version="3" description="Calling
Line Identification Restriction" active="false" enabled="false">
        <msd>

```

```

        <property name="enumMode" type="string"
description="Internal field to validate mode property" value="user_defined|restrict_all"/>
        </msd>
        <csd>
            <property name="mode" type="enum"
description="The mode of the working clir service (user_defined|restrict_all)"
value="restrict_all"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance" xs:noNamespaceSchemaLocation="ss.xsd" name="clip" version="5" description="Calling
Line Identification Presentation" active="false" enabled="false"/>
        <ss
            xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="click_to_call" version="5"
description="Click to call" active="false" enabled="false">
                <csd>
                    <property name="number" type="address"
description="User's number"/>
                    <property name="auto_answer"
type="boolean" description="Auto answer" value="true"/>
                    <property_list name="hashes"
type="string" maxlen="10" description="User's hashes">
                        <undefined/>
                    </property_list>
                </csd>
            </ss>
            <ss
                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="chunt" version="7"
description="Call hunt groups" active="false" enabled="false">
                    <msd>
                        <property name="regimeMode"
type="string" description="Internal field to validate search_strategy property"
value="group:numbers:queue_strategy|
serial:numbers:cyclic:search_strategy>window_shift_timeout:max_shift_count:queue_strategy|
longest_idle:numbers:cyclic>window_shift_timeout:max_shift_count:queue_strategy|
delay_group:numbers>window_shift_timeout:queue_strategy|manual"/>
                            <property name="enumSearch_strategy"
type="string" description="Internal field to validate search_strategy property" value="first|
last|random|longest_idle"/>
                            <property name="enumQueue_strategy"
type="string" description="Internal field to validate queue_strategy property" value="wait|
drop"/>
                            <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                <item value="assistant"/>
                                <item value="follow_me"/>
                                <item value="cfu_type2"/>
                            </property_list>
                    </msd>
                </ss>
            </ss>
        </ss>
    </ss>

```

```

        <item value="meet_me_conference"/>
        <item value="cfu"/>
        <item value="find_me"/>
        <item value="cf_aon"/>
        <item value="cft"/>
        <item value="cgg"/>
    </property_list>
</msd>
<ssd>
    <property name="delay_retrieve_timeout
_1" type="integer" description="Delay retrieve timeout for release group call by group
subscriber" value="1000"/>
    <property name="delay_retrieve_timeout
_2" type="integer" description="Delay retrieve timeout for release group call by slave
subscriber" value="5000"/>
    <property name="delay_retrieve_timeout
_3" type="integer" description="Delay retrieve timeout for waiting free group subscriber"
value="15000"/>
    <property name="forwarding_ss"
type="boolean" description="" value="true"/>
</ssd>
<csd>
    <property name="mode" type="regime"
description="The type of the chunt"/>
    <property name="cyclic" type="boolean"
description="Flag that is show is call hunt group cyclic or not" value="false"/>
    <property name="reset_window_start_pos
_after_call" type="boolean" description="Reset window start position after all incoming calls
will be connected with group calls" value="false"/>
    <property name="window_shift_timeout"
type="positive_integer" description="Interval after which window shift on the window_shift
value" value="0"/>
    <property name="max_shift_count"
type="integer" description="The number of times that a window can be shift" value="-1"/>
    <property name="window_size"
type="positive_integer" description="The number of simultaneous calls to the call-hunt's
subscribers" value="1"/>
    <property name="window_start_pos"
type="integer" description="The index (zero-based) of the first subscriber in the window"
value="0"/>
    <property name="window_shift"
type="positive_integer" description="The value on which shift window_start_pos when
window_shift_timeout fired" value="1"/>
    <property name="queue_strategy"
type="enum" description="(wait | drop)" value="wait"/>
    <property name="search_strategy"
type="enum" description="(first | last | random | longest_idle)" value="first"/>
    <property name="queue_size"
type="positive_integer" description="Wait calls' queue size" value="0"/>

```



```

                                <property_list name="numbers"
type="address" maxlen="10" description="Call-hunt member list">
                                <undefined/>
                                </property_list>
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="chold" version="8"
                                description="Call hold" active="false" enabled="false">
                                <ssd>
                                <property name="dtmf_detector"
type="boolean" description="" value="true"/>
                                </ssd>
                                <csd>
                                <property name="dtmf_sequence_as_flash
" type="boolean" description="Try interpret the DTMF sequence as a flash (look at incall
feature codes)" value="false"/>
                                <property name="disable_moh"
type="boolean" description="Disable music on hold" value="false"/>
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cgg" version="7"
                                description="Group-calling" active="false" enabled="false">
                                <msd>
                                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                    <item value="assistant"/>
                                    <item value="follow_me"/>
                                    <item value="cfu_type2"/>
                                    <item value="meet_me_conference"/>
                                    <item value="cfu"/>
                                    <item value="find_me"/>
                                    <item value="cf_aon"/>
                                    <item value="cft"/>
                                </property_list>
                                </msd>
                                <ssd>
                                <property name="forwarding_ss"
type="boolean" description="" value="true"/>
                                </ssd>
                                <csd>
                                <property name="send_call_diversion"
type="boolean" description="Send call diversion flag in call signalisation" value="true"/>
                                <property_list name="numbers"
type="address" maxlen="10" description="Group's numbers">
                                    <undefined/>
                                </property_list>

```

```

        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfu_type2" version="14"
        description="Call Forwarding Unconditional without notification" active="false"
        enabled="false">
        <msd>
            <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
            <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                <item value="assistant"/>
                <item value="follow_me"/>
            </property_list>
        </msd>
        <ssd>
            <property name="forwarding_ss"
type="boolean" description="" value="true"/>
        </ssd>
        <csd>
            <property name="number" type="address"
description="Forward number"/>
            <property name="use_redirection_counte
r" type="boolean" description="Use redirectionCounter field to count hops" value="false"/>
            <property name="external_number"
type="address" description="External forward number" value=""/>
            <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
        </csd>
    </ss>
    <ss
        xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfu" version="14"
        description="Call Forwarding Unconditional" active="false" enabled="false">
        <msd>
            <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
            <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                <item value="assistant"/>
                <item value="follow_me"/>
                <item value="cfu_type2"/>
                <item value="meet_me_conference"/>
            </property_list>
        </msd>
        <ssd>
            <property name="forwarding_ss"
type="boolean" description="" value="true"/>
        </ssd>

```

```

                                <csd>
                                <property name="number" type="address"
description="Forward number"/>
                                <property name="external_number"
type="address" description="External forward number" value=""/>
                                <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cft" version="14"
description="Call Forwarding Unconditional Time Dependent" active="false" enabled="false">
                                <msd>
                                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
                                <property name="enumTime_format"
type="string" description="Internal field to validate time_format property" value="local|utc"/
>
                                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                <item value="assistant"/>
                                <item value="follow_me"/>
                                <item value="cfu_type2"/>
                                <item value="meet_me_conference"/>
                                <item value="cfu"/>
                                <item value="find_me"/>
                                <item value="cf_aon"/>
                                </property_list>
                                </msd>
                                <ssd>
                                <property name="forwarding_ss"
type="boolean" description="" value="true"/>
                                </ssd>
                                <csd>
                                <property name="day_1" type="term"
description="{[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                <property name="day_2" type="term"
description="{[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                <property name="day_3" type="term"
description="{[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                <property name="day_4" type="term"
description="{[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>
                                <property name="day_5" type="term"
description="{[{{FromHour,FromMinute}, {ToHour,ToMinute}}, ...], Number, ExternalNumber}"
value="off"/>

```

```

                                <property name="day_6" type="term"
description="{[{FromHour,FromMinute}, {ToHour,ToMinute}], ..., Number, ExternalNumber}"
value="off"/>
                                <property name="day_7" type="term"
description="{[{FromHour,FromMinute}, {ToHour,ToMinute}], ..., Number, ExternalNumber}"
value="off"/>
                                <property name="time_format"
type="enum" description="CFT for Sunday (local | utc)" value="local"/>
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfos" version="12"
description="Call Forwarding Out of Service" active="false" enabled="false">
                                <msd>
                                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
                                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                <item value="cfu"/>
                                <item value="assistant"/>
                                <item value="follow_me"/>
                                <item value="tsmn_request"/>
                                </property_list>
                                </msd>
                                <ssd>
                                <property name="forwarding_ss"
type="boolean" description="" value="true"/>
                                </ssd>
                                </csd>
                                <property name="number" type="address"
description="Forward number"/>
                                <property name="external_number"
type="address" description="External forward number" value=""/>
                                <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfnr_type2" version="11"
description="Call Forwarding No Reply without notification" active="false" enabled="false">
                                <msd>
                                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
                                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                <item value="cfu"/>
                                <item value="assistant"/>
                                <item value="cfnr"/>

```

```

        <item value="follow_me"/>
        <item value="tsmn_request"/>
    </property_list>
</msd>
<ssd>
    <property name="forwarding_ss"
type="boolean" description="" value="true"/>
    </ssd>
<csd>
    <property name="number" type="address"
description="Forward number"/>
    <property name="timeout"
type="positive_integer" description="Timeout before call forward (sec)" value="15"/>
    <property name="use_redirection_counter" type="boolean" description="Use redirectionCounter field to count hops" value="false"/>
    <property name="external_number"
type="address" description="External forward number" value=""/>
    <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
    </csd>
</ss>
<ss
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfnr" version="11"
description="Call Forwarding No Reply" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
        <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
            <item value="cfu"/>
            <item value="assistant"/>
            <item value="follow_me"/>
            <item value="tsmn_request"/>
        </property_list>
    </msd>
    <ssd>
        <property name="forwarding_ss"
type="boolean" description="" value="true"/>
    </ssd>
<csd>
    <property name="number" type="address"
description="Forward number"/>
    <property name="external_number"
type="address" description="External forward number" value=""/>
    <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
    <property name="timeout"
type="positive_integer" description="Timeout before call forward (sec)" value="15"/>
    </csd>

```

```

        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cfb" version="11"
description="Call Forwarding Busy" active="false" enabled="false">
            <msd>
                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                    <item value="cfu"/>
                    <item value="assistant"/>
                    <item value="follow_me"/>
                    <item value="tsmn_request"/>
                </property_list>
            </msd>
            <ssd>
                <property name="forwarding_ss"
type="boolean" description="" value="true"/>
            </ssd>
            <csd>
                <property name="number" type="address"
description="Forward number"/>
                <property name="external_number"
type="address" description="External forward number" value=""/>
                <property name="use_external_number"
type="boolean" description="Use external forward number" value="false"/>
            </csd>
        </ss>
        <ss
            xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cf_sip" version="9"
description="Call Forwarding by SIP redirection (302 message)" active="false" enabled="false">
            <msd>
                <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
                <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                    <item value="cfu"/>
                    <item value="assistant"/>
                    <item value="follow_me"/>
                    <item value="tsmn_request"/>
                </property_list>
            </msd>
            <ssd>
                <property name="forwarding_ss"
type="boolean" description="" value="true"/>
            </ssd>
        </ss>
    </ss>

```

```

xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cf_aon" version="9"
description="Call Forwarding AON" active="false" enabled="false">
  <msd>
    <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
    <property_list name="conflict_list"
type="atom" maxlen="0" description="List of SS that is conflict with current SS">
      <item value="assistant"/>
      <item value="follow_me"/>
      <item value="cfu_type2"/>
      <item value="meet_me_conference"/>
      <item value="cfu"/>
      <item value="find_me"/>
    </property_list>
  </msd>
  <ssd>
    <property name="forwarding_ss"
type="boolean" description="" value="true"/>
  </ssd>
  <csd>
    <property name="no_reply"
type="boolean" description="Activate service when subscriber no reply" value="false"/>
    <property name="timeout"
type="positive_integer" description="Timeout before call forward (sec)" value="15"/>
    <property name="number" type="address"
description="Forward number"/>
    <property_list name="prefix_number"
type="address" maxlen="10" description="Prefix number A">
      <undefined/>
    </property_list>
  </csd>
</ss>
<ss
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="cc_agent" version="2"
description="Contact-Center agent" active="false" enabled="false"/>
  <ss
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="callback" version="5"
description="CallBack" active="false" enabled="false">
    <ssd>
      <property name="collect_digits_tim
eout" type="integer" description="wait timeout to collect cancel digits" value="60"/>
      <property name="a_onhook_timeout"
type="positive_integer" description="Timeout for A subscriber onhook (sec)" value="5"/>
      <property name="max_call_retry"
type="positive_integer" description="The maximum count of call's retry to initial subscriber"
value="5"/>
    </ssd>
  </ss>
</ss>

```

```

        <property name="dtmf_detector"
type="boolean" description="" value="true"/>
    </ssd>
</ss>
<ss
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="call_recording" version="11"
description="Call recording" active="false" enabled="false">
    <msd>
        <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="any"/>
        <property name="enumMode"
type="string" description="Internal field to validate mode property" value="always_on|
on_demand|after_answer"/>
    </msd>
    <ssd>
        <property name="locked"
type="boolean" description="Can subscriber manage current SS, or not" value="true"/>
        <property name="dtmf_detector"
type="boolean" description="" value="true"/>
    </ssd>
    <csd>
        <property name="mode" type="enum"
description="The mode of the recording call (always_on|on_demand)"/>
        <property name="separated"
type="boolean" description="The separated recording of the call. If true, the outgoing audio
of the subscriber is recorded to first audio track, the incoming audio of the subscriber is
recorded to second audio track. If false, the incoming and the outgoing audio of the
subscriber is recorded to first audio track." value="false"/>
    </csd>
</ss>
<ss
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="boss_group" version="1"
description="Boss-group" active="false" enabled="false">
    <msd>
        <property name="integer_rangeDelay
_ring_timeout" type="string" description="Internal field" value="[0,60]"/>
        <property name="integer_rangeLines
_count" type="string" description="Internal field" value="[0,16]"/>
        <property_list name="conflict_list
" type="atom" maxLen="0" description="List of SS that is conflict with current SS">
            <item value="cw"/>
        </property_list>
        <property_list name="dependency_li
st" type="atom" maxLen="0" description="List of SS that is depended from SS">
            <item value="auto_redial"/>
        </property_list>
    </msd>
</ssd>

```



```

type="boolean" description="" value="true"/>
                                </property name="dtmf_detector"
                                </ssd>
                                <csd>
                                <property name="display_name"
                                type="string" description="Display name for boss group" value=""/>
                                <property name="delay_ring_timeout
" type="integer_range" description="Timeout before call on delayed members" value="0"/>
                                <property name="lines_count"
                                type="integer_range" description="Incoming boss-group lines count" value="2"/>
                                <property name="members_fwd_busy_o
verride" type="boolean" description="Ignore members CFB for group calls" value="true"/>
                                <property name="call_waiting_tone"
                                type="boolean" description="Play call waiting tone for second(s) calls on group"
                                value="false"/>
                                <property_list name="members"
                                type="address" maxlen="8" description="Members for immediate call">
                                    <undefined/>
                                </property_list>
                                <property_list name="delay_members
" type="address" maxlen="8" description="Members for delay call">
                                    <undefined/>
                                </property_list>
                                <property_list name="silent_member
s" type="address" maxlen="8" description="Members for BLF and pickup only group calls">
                                    <undefined/>
                                </property_list>
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="blf" version="3"
                                description="Busy Lamp Field" active="false" enabled="false">
                                <csd>
                                <property name="max_subscribe"
                                type="positive_integer" description="Maximum number of subscriptions" value="16"/>
                                <property name="ignore_page_calls"
                                type="boolean" description="Doesn't send BLF notification from Zone Page calls" value="true"/
                                >
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="avatar" version="2"
                                description="Picture for outgoing calls" active="false" enabled="false">
                                <csd>
                                <property name="pic_addr"
                                type="uri_string" description="URL to access the image" value=""/>
                                </csd>
                                </ss>

```

```

<ss
  xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="auto_redial_with_callback"
  version="5" description="Auto redial with callback" active="false" enabled="false">
  <msd>
    <property name="enumRingback"
type="string" description="Ringback" value="tone|moh"/>
  </msd>
  <csd>
    <property name="recall_timer"
type="positive_integer" description="Time between recalls (sec)" value="90"/>
    <property name="recall_count"
type="positive_integer" description="Recalls' count" value="30"/>
    <property name="a_no_answer_timeou
t" type="positive_integer" description="Awaiting B-side answer timeout (sec)" value="10"/>
    <property name="b_no_answer_timeou
t" type="positive_integer" description="Awaiting A-side answer timeout (sec)" value="30"/>
    <property name="ringback"
type="enum" description="Ringback (tone | moh)" value="tone"/>
  </csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="auto_redial" version="2"
  description="Auto redial" active="false" enabled="false">
  <csd>
    <property name="recall_timeout"
type="positive_integer" description="Timeout between recalls (sec)" value="15"/>
    <property name="recall_count"
type="positive_integer" description="Recalls' count" value="6"/>
    <property name="no_answer_timeout"
type="positive_integer" description="Awaiting answer timeout (sec)" value="30"/>
  </csd>
</ss>
<ss
  xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="assistant" version="2"
  description="Chief-assistant group" active="false" enabled="false">
  <msd>
    <property name="owner" type="atom"
description="SS type. Available values: user | trunk | any" value="user"/>
  </msd>
  <ssd>
    <property name="forwarding_ss"
type="boolean" description="" value="true"/>
  </ssd>
  <csd>
    <property_list name="assistants"
type="index_address" maxlen="4" description="Assistants numbers. Format: <address>[/
<index>]">

```

```

        <undefined/>
    </property_list>
    <property_list name="whitelist"
type="index_address" maxlen="128" description="White list members. Format: <address>[/
<index>]">
        <undefined/>
    </property_list>
</csd>
</ss>
<ss
xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="alarm" version="8"
description="Alarm Call" active="false" enabled="false">
    <msd>
        <property name="handle_module"
type="atom" description="" value="ss_alarm_handler"/>
    </msd>
    <ssd>
        <property name="collect_digits_timeo
out" type="integer" description="Wait timeout to collect cancel digits" value="60"/>
    </ssd>
    <csd>
        <property name="no_answer_timeout"
type="integer" description="Awaiting answer timer (value in seconds)" value="45"/>
        <property name="call_attempt_count
" type="integer" description="The count of the call's attempt" value="3"/>
        <property name="call_attempt_timeo
ut" type="integer" description="Awaiting timer between calls' attempts" value="60"/>
        <property name="cancel_digits"
type="address" description="Digits to cancel alarm" value="*"/>
        <property name="alarm_1"
type="term" description="{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "],{"HH, MM"}[,
AlarmName]}" value="off"/>
        <property name="alarm_2"
type="term" description="{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "],{"HH, MM"}[,
AlarmName]}" value="off"/>
        <property name="alarm_3"
type="term" description="{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "],{"HH, MM"}[,
AlarmName]}" value="off"/>
        <property name="alarm_4"
type="term" description="{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "],{"HH, MM"}[,
AlarmName]}" value="off"/>
        <property name="alarm_5"
type="term" description="{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "],{"HH, MM"}[,
AlarmName]}" value="off"/>
        <property name="alarm_6"
type="term" description="{" "'one-time'" | daily, "[" Day1[, Day2][, ...] "],{"HH, MM"}[,
AlarmName]}" value="off"/>

```

```

                                <property name="alarm_7"
type="term" description="{ " 'one-time' | daily, [" Day1[, Day2][, ...] "], {"HH, MM"}[,
AlarmName]}"" value="off"/>
                                <property name="alarm_8"
type="term" description="{ " 'one-time' | daily, [" Day1[, Day2][, ...] "], {"HH, MM"}[,
AlarmName]}"" value="off"/>
                                <property name="alarm_9"
type="term" description="{ " 'one-time' | daily, [" Day1[, Day2][, ...] "], {"HH, MM"}[,
AlarmName]}"" value="off"/>
                                </csd>
                                </ss>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="acb" version="2"
description="Anonymous Call Blocking" active="false" enabled="false"/>
                                <ss
                                xmlns:xs="http://www.w3.org/2001/
XMLSchema-instance" xs:noNamespaceSchemaLocation="ss.xsd" name="3way" version="6"
description="3-Way Conference" active="false" enabled="false">
                                <msd>
                                <property_list name="dependenc
y_list" type="atom" maxlen="0" description="List of SS that is depended from SS">
                                <item value="chold"/>
                                </property_list>
                                <property_list name="conflict_
list" type="atom" maxlen="0" description="List of SS that is conflict with current SS">
                                <item value="conference"/>
                                </property_list>
                                </msd>
                                <ssd>
                                <property name="dtmf_detector"
type="boolean" description="" value="true"/>
                                </ssd>
                                </ss>
                                </sss>
                                </out>

```

Нс_ss_uninstall

Команда позволяет удалить услуги из системы.

При этом команда удаляет все настройки данной услуги у всех абонентов, которые его использовали.

Аналог команды в CoCon:

```
/cluster/storage/<CLUSTER>/ss/uninstall
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/ss_uninstall

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="storage" type="xs:string" use="required"/>
            <xs:attribute name="name" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/ss_uninstall

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ss_uninstall.xsd">
  <request storage="ds1" name="mcid"/>
</in>
```

Ответ - 204:

Пустой xml.

Hc_statistics_show - Управление статистикой ECSS-10

Hc_statistics_show

Команда предназначена для получения статистики ECSS-10 за определенный интервал времени с использованием определенных фильтров.

Аналог команды в CoCon:

`/domain/<DOMAIN>/core/statistics/show`

Метод http запроса:

POST

Шаблон http запроса:

`http://host:port/commands/ecss_statistics_show`

Код http ответа:

201 – в случае успеха;

404 – в случае ошибки.

Xsd схема xml файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="common.xsd"/>
  <!-- systemType -->
  <xs:simpleType name="systemType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="ecss10.common"/>
      <xs:enumeration value="ecss10.ssw"/>
      <xs:enumeration value="ecss10.cc"/>
      <xs:enumeration value="ecss10.tc"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- systemType -->
  <xs:simpleType name="measureType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="realtime"/>
      <xs:enumeration value="intrahour"/>
      <xs:enumeration value="hour"/>
      <xs:enumeration value="day"/>
      <xs:enumeration value="week"/>
      <xs:enumeration value="month"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- subsystemType -->
  <xs:simpleType name="subsystemType">
    <xs:restriction base="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          In ECSS-10 v. 3.4.2 are supported subsystems:
          queue, agent, core, sip, megaco, sigtran, ds, mediator, mycelium,
          trunk_manager, direction_manager.
        </xs:documentation>
      </xs:annotation>
    </xs:restriction>
  </xs:simpleType>
  <!-- statisticType -->
  <xs:simpleType name="statisticType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="counter"/>
      <xs:enumeration value="gauge"/>
      <xs:enumeration value="meter"/>
      <xs:enumeration value="histogram"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- statisticsValueType -->
  <xs:complexType name="statisticsValueType">
    <xs:attribute name="time" type="dateTimeType"/>
    <xs:attribute name="value" type="xs:int"/>
  </xs:complexType>

```

```

    <xs:attribute name="n" type="xs:int"/>
    <xs:attribute name="min" type="xs:int" use="optional"/>
    <xs:attribute name="max" type="xs:int" use="optional"/>
    <xs:attribute name="avg" type="xs:double" use="optional"/>
  </xs:complexType>
  <!-- statisticsInType -->
  <xs:complexType name="statisticsInType">
    <xs:attribute name="from" type="dateTimeType"/>
    <xs:attribute name="to" type="dateTimeType"/>
    <xs:attribute name="system" type="systemType"/>
    <xs:attribute name="subsystem" type="subsystemType" use="optional"/>
    <xs:attribute name="statistic" type="xs:string" use="optional"/>
    <xs:attribute name="type" type="statisticTypeType" use="optional"/>
    <xs:attribute name="variable_part_1" type="xs:string" use="optional"/>
    <xs:attribute name="variable_part_2" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- statisticsOutType -->
  <xs:complexType name="statisticsOutType">
    <xs:sequence>
      <xs:element name="value" minOccurs="0" maxOccurs="unbounded"
type="statisticsValueType"/>
    </xs:sequence>
    <xs:attribute name="system" type="systemType"/>
    <xs:attribute name="subsystem" type="subsystemType"/>
    <xs:attribute name="statistic" type="xs:string"/>
    <xs:attribute name="type" type="statisticTypeType"/>
    <xs:attribute name="measure" type="measureType" use="optional"/>
    <xs:attribute name="variable_part_1" type="xs:string" use="optional"/>
    <xs:attribute name="variable_part_2" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="statistics" maxOccurs="unbounded"
type="statisticsInType"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>

```



```

        <xs:element name="response">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="statistics" minOccurs="0" maxOccurs="unbounded"
type="statisticsOutType"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример 1

Запрос: http://192.168.1.21:9999/commands/ecss_statistics_show

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ecss_statistics_show.xsd">
  <request domain="biysk.local">
    <statistics system="ecss10.ssw" subsystem="trunk_manager" from="2017/12/05 00:00:00"
to="2017/12/10 00:00:00" statistic="incoming_seizures.counter"/>
  </request>
</in>

```

Ответ:

Код: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ecss_statistics_show.xsd">
  <response>
    <statistics system="ecss10.ssw" subsystem="trunk_manager"
  statistic="incoming_seizures.counter" measure="day" type="counter" variable_part_1="smg-4">
      <value time="2017/12/05 16:59:59" value="7" n="" min="" max="" avg=""/>
      <value time="2017/12/06 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/07 16:59:59" value="21" n="" min="" max="" avg=""/>
      <value time="2017/12/08 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/09 16:59:59" value="13" n="" min="" max="" avg=""/>
    </statistics>
    <statistics system="ecss10.ssw" subsystem="trunk_manager"
  statistic="incoming_seizures.counter" measure="day" type="counter"
  variable_part_1="system:teleconference">
      <value time="2017/12/05 16:59:59" value="1" n="" min="" max="" avg=""/>
      <value time="2017/12/06 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/07 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/08 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/09 16:59:59" value="2" n="" min="" max="" avg=""/>
    </statistics>
  </response>
</out>

```

Пример 2

Запрос: http://192.168.23.34:9999/commands/ecss_statistics_show

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ecss_statistics_show.xsd">
  <request domain="biysk.local">
    <statistics system="ecss10.ssw" subsystem="trunk_manager" from="2017/12/05 00:00:00"
  to="2017/12/10 00:00:00" statistic="incoming_seizures.counter" />
    <statistics system="ecss10.ssw" subsystem="trunk_manager" from="2017/12/05 00:00:00"
  to="2017/12/10 00:00:00" statistic="incoming_accepted_seizures.counter" />
  </request>
</in>

```

Ответ:

Код: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="ecss_statistics_show.xsd">
  <response>
    <statistics system="ecss10.ssw" subsystem="trunk_manager"
  statistic="incoming_seizures.counter" measure="day" type="counter" variable_part_1="smg-4">
      <value time="2017/12/05 16:59:59" value="7" n="" min="" max="" avg=""/>
      <value time="2017/12/06 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/07 16:59:59" value="21" n="" min="" max="" avg=""/>
      <value time="2017/12/08 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/09 16:59:59" value="13" n="" min="" max="" avg=""/>
    </statistics>
    <statistics system="ecss10.ssw" subsystem="trunk_manager"
  statistic="incoming_seizures.counter" measure="day" type="counter"
  variable_part_1="system:teleconference">
      <value time="2017/12/05 16:59:59" value="1" n="" min="" max="" avg=""/>
      <value time="2017/12/06 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/07 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/08 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/09 16:59:59" value="2" n="" min="" max="" avg=""/>
    </statistics>
    <statistics system="ecss10.ssw" subsystem="trunk_manager"
  statistic="incoming_accepted_seizures.counter" measure="day" type="counter"
  variable_part_1="smg-4">
      <value time="2017/12/05 16:59:59" value="7" n="" min="" max="" avg=""/>
      <value time="2017/12/06 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/07 16:59:59" value="21" n="" min="" max="" avg=""/>
      <value time="2017/12/08 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/09 16:59:59" value="13" n="" min="" max="" avg=""/>
    </statistics>
    <statistics system="ecss10.ssw" subsystem="trunk_manager"
  statistic="incoming_accepted_seizures.counter" measure="day" type="counter"
  variable_part_1="system:teleconference">
      <value time="2017/12/05 16:59:59" value="1" n="" min="" max="" avg=""/>
      <value time="2017/12/06 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/07 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/08 16:59:59" value="" n="" min="" max="" avg=""/>
      <value time="2017/12/09 16:59:59" value="2" n="" min="" max="" avg=""/>
    </statistics>
  </response>
</out>

```

Hc_sudo - Привилегированный режим

- [Hc_sudo_cmd](#)
- [Hc_unsudo_cmd](#)

Ис_sudo_cmd

Команда позволяет выполнить команду с правами пользователя root.

Аналог команды в CoCon:

`sudo <cmd>`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/sudo_cmd`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 401 - параметры авторизации введены неверно;
- другие коды ответа зависят от команды, которая выполняется из под sudo.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="args" type="xs:base64Binary"/>
            </xs:sequence>
            <xs:attribute name="command" type="xs:string" use="required"/>
            <xs:attribute name="tag" type="xs:string" use="required"/>
            <xs:attribute name="user" type="xs:string" use="optional"/>
            <xs:attribute name="header" type="xs:string" use="optional"/>
            <xs:attribute name="passwd_prompt" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/sudo_cmd

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_sudo.xsd">
  <request header="Sudo required" passwd_prompt="Enter password:" tag="cluster-manager"
user="ssw" command="list_of_clusters">
    <args>PGluIHhtbG5zOnhzaT0iaHR0cDovL3d3dy53My5vcmcvMjAwMS9YTUxTY2h1bWETAw5zdGFuY2Ui
IHhzaTpub05hbWVzcGFjZVNjaGVtYUxvY2F0aw9uPSJoY19saXN0X29mX2NsdXN0ZXJzLnhzZCI+
ICAgICAgICAg8cmVxdwVzdCBzdG9yYWdlPSJKcyIvPiAgICAgICAgPC9pbj4=</args>
  </request>
</in>
```

Ответ:

- в случае неверной авторизации: 401
- в случае успеха: 200

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_list_of_clusters.xsd">
  <clusters>
    <cluster name="tts" role="tts" active="true"/>
    <cluster name="core1" role="core" active="true"/>
    <cluster name="bot" role="adapter" active="true"/>
    <cluster name=".virtual" role="adapter" active="true"/>
    <cluster name="md" role="mediator" active="true"/>
    <cluster name="ds_test" role="storage" active="true"/>
    <cluster name="core2" role="core" active="true"/>
    <cluster name=".system_bridge" role="adapter" active="true"/>
    <cluster name="core3" role="core" active="true"/>
    <cluster name="pa_sip" role="adapter" active="true"/>
    <cluster name="ds" role="storage" active="true"/>
    <cluster name="megaco1" role="adapter" active="true"/>
  </clusters>
</out>
```

hc_unsudo_cmd

Команда позволяет закрыть sudo-сессия для определенного пользователя с определенным тегом.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/unsudo_cmd

Код HTTP-ответа:

- 201 - в случае, если sudo-сессия успешно закрылась;

- 404 - в противном случае.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="tag" type="xs:string" use="required"/>
            <xs:attribute name="user" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример::

Запрос: http://192.168.1.21:9999/commands/unsudo_cmd

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_unsudo.xsd">
  <request tag="cluster-manager" user="admin"/>
</in>
```

Ответ в случае успеха: 204

Hc_sys_iface_options - Управление настройками контекста маршрутизации для системных интерфейсов

- [Hc_sys_iface_options_clean](#)
- [Hc_sys_iface_options_info](#)
- [Hc_sys_iface_options_set](#)

[Hc_sys_iface_options_clean](#)

Команда сброса настроек контекста маршрутизации для системных интерфейсов.

Аналог команды в CoCon:

`domain/<DOMAIN>/system-iface/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sys_iface_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean properties of system iface params
      Usage: http://server:9999/commands/sys_iface_options_clean.
      Implementation: hc_sys_iface_options_clean.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sys_iface_options_clean


```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sys_iface_options_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="routing.context"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sys_iface_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="routing.context"/>
      </domain>
    </domain_options>
  </result>
</out>

```

hc_sys_iface_options_info

Команда просмотра информации о контексте маршрутизации для системных интерфейсов.

Аналог команды в CoCon:

domain/<DOMAIN>/system-iface/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sys_iface_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show properties of iface user modifications
      Usage: http://server:9999/commands/sys_iface_options_info.
      Implementation: hc_sys_iface_options_info.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="ifaceUserOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="routing.context" type="xs:string" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType -->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainInfoOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainInfoInType" minOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/sys_iface_options_info

```

<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_sys_iface_options_info.xsd">
  <request>
    <domain_options domain="biysk.local"/>
  </request>
</in>

```

Ответ:200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_ftp_domain_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="ttsFtpOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="ttsFtpOptionsType" password="cdr"/>
      </domain>
    </domain_options>
  </result>
</out>

```

hc_sys_iface_options_set

Команда изменения настроек контекста маршрутизации для системных интерфейсов.

Аналог команды в CoCon:

domain/<DOMAIN>/system-iface/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/sys_iface_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set system iface options
      Usage: http://server:9999/commands/sys_iface_options_set
      Implementation: hc_sys_iface_options_set.erl.
      Git: git@git.eltex.loc:httpTerminal.git.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <xs:complexType name="ifaceUserOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="routing.context" type="xs:string" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainSetOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType" minOccurs="1" maxOccurs="1"/>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1,21:9999/commands/sys_iface_options_set

```

<?xml version="1.0"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sys_iface_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="ifaceUserOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="ifaceUserOptionsType"
routing.context="system:teleconference=ctx_from_local;system:ivr=ctx_from_local"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_sys_iface_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="routing.context"/>
        <success option="routing.context"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Hc_system_geo_options - Управление настройками ГЕО-резерва

- [Hc_system_geo_options_clean](#)
- [Hc_system_geo_options_info](#)
- [Hc_system_geo_options_set](#)

[Hc_system_geo_options_clean](#)

Команда сброса настроек ГЕО-резерва.

Аналог команды в CoCon:

system/geo-backup/properties/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/system_geo_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean system geo-backup options.
      Usage: http://server:9999/commands/system\_geo\_options\_clean.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- geoSystemOptionsType -->
  <xs:complexType name="geoSystemOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="slave_sites" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              usage:
              site1;site2;site3
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="enabled" type="booleanType"/>
        <xs:attribute name="site" type="stringType"/>
        <xs:attribute name="sync_timeout" type="integerType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>

```

```

        <xs:element name="result">
            <xs:complexType>
                <xs:all>
                    <xs:element name="system_options" type="systemCleanOutType"/>
                </xs:all>
            </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" />
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/system_geo_options_clean

```

<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="system_geo_options_set.xsd">
    <request>
        <system_options>
            <system>
                <option name="enabled"/>
                <option name="site"/>
                <option name="slave_sites"/>
                <option name="sync_timeout"/>
            </system>
        </system_options>
    </request>
</in>

```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="system_geo_options_clean.xsd">
  <result>
    <system_options>
      <system>
        <success option="sync_timeout"/>
        <success option="slave_sites"/>
        <success option="site"/>
        <success option="enabled"/>
      </system>
    </system_options>
  </result>
</out>
```

Нс_system_geo_options_info

Команда просмотра настроек ГЕО-резерва.

Аналог команды в CoCon:

system/geo-backup/properties/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/system_geo_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show system geo-backup options.
      Usage: http://server:9999/commands/system\_geo\_options\_info.
    </xs:documentation>
  </xs:annotation>
</xs:annotation>
<xs:include schemaLocation="error.xsd"/>
<xs:include schemaLocation="ecm_common.xsd"/>

<!-- geoSystemOptionsType -->
<xs:complexType name="geoSystemOptionsType">
  <xs:complexContent>
    <xs:extension base="baseOptionsType">
      <xs:attribute name="slave_sites" type="stringType">
        <xs:annotation>
          <xs:documentation xml:lang="en">
            usage:
              site1;site2;site3
          </xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="enabled" type="booleanType"/>
      <xs:attribute name="site" type="stringType"/>
      <xs:attribute name="sync_timeout" type="integerType"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="system_options" type="systemInfoInType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>

```

```

        <xs:element name="result">
            <xs:complexType>
                <xs:all>
                    <xs:element name="system_options" type="systemInfoOutType"/>
                </xs:all>
            </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/system_geo_options_info

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="system_geo_options_info.xsd">
    <request>
        <system_options/>
    </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
    xs:noNamespaceSchemaLocation="system_geo_options_info.xsd">
    <result>
        <system_options>
            <system>
                <options xs:type="geoSystemOptionsType" enabled="false" site="ecss10"
                slave_sites="" sync_timeout="3600000"/>
            </system>
        </system_options>
    </result>
</out>

```

Нс_system_geo_options_set

Команда изменения настроек ГЕО-резерва.

Аналог команды в CoCon:

system/geo-backup/properties/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/system_geo_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set system geo-backup options.
      Usage: http://server:9999/commands/geo\_system\_options\_set.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- geoSystemOptionsType -->
  <xs:complexType name="geoSystemOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="slave_sites" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              usage:
                site1;site2;site3
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="enabled" type="booleanType"/>
        <xs:attribute name="site" type="stringType"/>
        <xs:attribute name="sync_timeout" type="integerType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="system_options" type="systemSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>

```

```

        <xs:element name="result">
            <xs:complexType>
                <xs:all>
                    <xs:element name="system_options" type="systemSetOutType"/>
                </xs:all>
            </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" />
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/system_geo_options_set

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="system_geo_options_set.xsd">
    <request>
        <system_options>
            <system>
                <options enabled="false" site="ecss10" slave_sites="site1;site2"
sync_timeout="1000000" xs:type="geoSystemOptionsType"/>
            </system>
        </system_options>
    </request>
</in>

```

Ответ: 200


```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="system_geo_options_set.xsd">
  <result>
    <system_options>
      <system>
        <success option="sync_timeout"/>
        <success option="slave_sites"/>
        <success option="site"/>
        <success option="enabled"/>
      </system>
    </system_options>
  </result>
</out>
```

Hc_tc_meetings - Управление историей совещаний

- [Hc_tc_meetings_active_info](#)
- [Hc_tc_meetings_active_list](#)
- [Hc_tc_meetings_history](#)
- [Hc_tc_meetings_info](#)
- [Hc_tc_meetings_list](#)

[Hc_tc_meetings_active_info](#)

Команда предназначена для просмотра списка активных телеконференций.

Аналог команды в CoCon:

`/domain/<DOMAIN>/tc/meeting/active/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/tc_meetings_active_info

Код ответа:

200 - успешно;
404 - ошибка.

XSD-схема, валидирующая входные и выходные данные:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="common_types.xsd"/>
  <!-- Types -->
  <xs:complexType name="memberType">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="voice" type="xs:string" use="required"/>
    <xs:attribute name="state" type="xs:string" use="required"/>
    <xs:attribute name="master" type="xs:boolean" use="optional"/>
    <xs:attribute name="hold" type="xs:boolean" use="optional"/>
  </xs:complexType>

  <xs:complexType name="meetingType">
    <xs:sequence>
      <xs:element name="member" type="memberType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="meeting" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="meeting" type="meetingType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="response" type="responseType" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```
</xs:element>
</xs:schema>
```

Примеры:

Вывести информацию обо всех активных совещаниях:

http://192.168.1.21:9999/commands/tc_meetings_active_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_tc_meetings_active_info.xsd">
  <request domain="test"/>
</in>
```

Ответ: 200 Если есть активные конференции:

```
<?xml version="1.0"?>
<out xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_tc_meetings_active_info.xsd">
  <response>
    <meeting id="836216376F0" name="test-tc-1" description="description">
      <member id="801" voice="on" state="in_conf" master="true" hold="false"/>
      <member id="802" voice="off" state="online" master="false" hold="false"/>
      <member id="803" voice="off" state="online" master="false" hold="false"/>
    </meeting>
    <meeting id="836216376F1" name="test-tc-2" description="description">
      <member id="901" voice="on" state="in_conf" master="true" hold="false"/>
      <member id="902" voice="off" state="online" master="false" hold="false"/>
      <member id="903" voice="off" state="online" master="false" hold="false"/>
    </meeting>
  </response>
</out>
```

Запрос с указанием имени телеконференции:

http://192.168.23.34:9999/commands/tc_meetings_active_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_tc_meetings_active_info.xsd">
  <request domain="test" meeting="test-tc-1"/>
</in>
```

Ответ:200

```
<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_tc_meetings_active_info.xsd">
  <response/>
</out>
```

[Hc_tc_meetings_active_list](#)

Команда предназначена для просмотра списка активных телеконференций.

Аналог команды в CoCon:

`/domain/<DOMAIN>/tc/meeting/active/list`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/tc_meetings_active_list

Код ответа:

200 - успешно;

404 - ошибка.

XSD-схема, валидирующая входные и выходные данные:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- Types -->
  <xs:complexType name="meetingType">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="meeting" type="meetingType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" type="requestType" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="response" type="responseType" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Вывести лист всех активных совещаний:

http://192.168.1.21:9999/commands/tc_meetings_active_list

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_tc_meetings_active_list.xsd">
  <request domain="test"/>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_tc_meetings_active_list.xsd">
  <response>
    <meeting id="836216376F0" name="test-tc" description="description"/>
  </response>
</out>
```

[Hc_tc_meetings_history](#)

Команда предназначена для получения истории совещаний.

Аналог команды в CoCon:

`/domain/<DOMAIN>/tc/meeting/history`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/tc_meetings_history

Код ответа:

200 - успешно;

404 - ошибка.

XSD-схема, валидирующая входные и выходные данные:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- Types -->
  <xs:complexType name="memberType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="meetingType">
    <xs:sequence>
      <xs:element name="member" type="memberType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="optional"/>
    <xs:attribute name="meeting_id" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="master" type="xs:string" use="required"/>
    <xs:attribute name="start_time" type="xs:string" use="required"/>
    <xs:attribute name="stop_time" type="xs:string" use="required"/>
    <xs:attribute name="duration" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:simpleType name="orderByType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="id"/>
      <xs:enumeration value="name"/>
      <xs:enumeration value="start_time"/>
      <xs:enumeration value="stop_time"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="sortByType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="desc"/>
      <xs:enumeration value="asc"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="lastType">
    <xs:attribute name="limit" type="xs:integer" use="optional"/>
    <xs:attribute name="order_by" type="orderByType" use="optional"/>
    <xs:attribute name="sort_by" type="sortByType" use="optional"/>
  </xs:complexType>

  <xs:complexType name="infoType">
    <xs:attribute name="id" type="xs:integer" use="required"/>
  </xs:complexType>

```

```

<xs:complexType name="requestType">
  <xs:choice>
    <xs:element name="last" type="lastType"/>
    <xs:element name="info" type="infoType"/>
    <xs:element name="all"/>
  </xs:choice>
  <xs:attribute name="domain" type="xs:string" use="required"/>
</xs:complexType>

<xs:complexType name="responseType">
  <xs:choice>
    <xs:sequence>
      <xs:element name="meeting" type="meetingType" minOccurs="0" maxOccurs="unbounded"
/>
    </xs:sequence>
  </xs:choice>
</xs:complexType>

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request" type="requestType"/>
    </xs:all>
  </xs:complexType>
</xs:element>

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="response" type="responseType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Примеры:

Запрос информации о всех совещаниях:

http://192.168.1.21:9999/commands/tc_meetings_history


```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_tc_meetings_history.xsd">
  <request domain="biysk.local">
    <all/>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_tc_meetings_history.xsd">
  <response>
    <meeting id="1" meeting_id="8362572E53B" name="tc_240475" master="240475" start_time="2017/12/09 20:26:15" stop_time="2017/12/09 20:28:32" duration="0:02:16">
      <member name="471040"/>
      <member name="89236501973"/>
    </meeting>
    <meeting id="3" meeting_id="8362572E53B" name="tc_240475" master="240475" start_time="2017/12/09 20:46:35" stop_time="2017/12/09 20:49:05" duration="0:02:29">
      <member name="89236501973"/>
    </meeting>
    <meeting id="5" meeting_id="8362572E53B" name="tc_240475" master="undefined" start_time="1970/01/01 07:00:00" stop_time="2017/12/09 20:49:05" duration="420229:49:05"/>
  </response>
</out>
```

Запрос подробной информации о совещании 3:

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_tc_meetings_history.xsd">
  <request domain="biysk.local">
    <info id="3" />
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_tc_meetings_history.xsd">
  <response>
    <meeting name="tc_240475" meeting_id="8362572E53B" master="240475" start_time="2017/12/09 20:46:35" stop_time="2017/12/09 20:49:05" duration="0:02:29">
      <member name="89236501973"/>
    </meeting>
  </response>
</out>
```

[Hc_tc_meetings_info](#)

Команда предназначена для получения информации о совещаниях.

Аналог команды в CoCon:

domain/<DOMAIN>/tc/meeting/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/tc_meetings_info

Код ответа:

200 - успешно;

404 - ошибка.

XSD-схема, валидирующая входные и выходные данные:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="common_types.xsd"/>
  <!-- Types -->
  <xs:complexType name="memberType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="groupType">
    <xs:sequence>
      <xs:element name="member" type="memberType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:integer" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="masterType">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="read" type="xs:boolean" use="optional"/>
    <xs:attribute name="write" type="xs:boolean" use="optional"/>
  </xs:complexType>

  <xs:complexType name="meetingType">
    <xs:sequence>
      <xs:element name="group" type="groupType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="master" type="masterType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="greeting_url" type="xs:string" use="optional"/>
    <xs:attribute name="is_public" type="xs:boolean" use="required"/>
    <xs:attribute name="is_open" type="xs:boolean" use="required"/>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="meeting" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="meeting" type="meetingType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>

```

```
</xs:complexType>

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request" type="requestType"/>
    </xs:all>
  </xs:complexType>
</xs:element>

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="response" type="responseType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>
```

Пример:

Запрос:

http://192.168.1.21:9999/commands/tc_meetings_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="tc_meetings_info.xsd">
  <request domain="biysk.local"/>
</in>
```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_tc_meet
ings_info.xsd">
  <response>
    <meeting name="default" id="default" description="" greeting_url="" is_public="true"
is_open="false"/>
    <meeting name="new_meeting" id="83620F86F67" description="" greeting_url="" is_public="tr
ue" is_open="true">
      <group id="1" name="">
        <member name="240101">
          <property name="role" value="active"/>
        </member>
        <member name="240466">
          <property name="role" value="active"/>
        </member>
      </group>
    </meeting>
    <meeting name="ltp-3" id="83622413D" description="" greeting_url="system://sounds/
ai_conference_prompt.wav" is_public="true" is_open="true">
      <group id="1" name="">
        <member name="240101">
          <property name="role" value="active"/>
        </member>
        <member name="240462">
          <property name="role" value="active"/>
        </member>
        <member name="240464">
          <property name="role" value="passive"/>
        </member>
        <member name="416977">
          <property name="role" value="passive"/>
        </member>
      </group>
    </meeting>
    <meeting name="new_meeting2" id="83626ECEF82" description="мое новое совещание"
greeting_url="system://sounds/ai_participant_of_conference.wav" is_public="true" is_open="true">
      <group id="1" name="Group One">
        <member name="240502">
          <property name="role" value="passive"/>
        </member>
        <member name="240503">
          <property name="role" value="active"/>
        </member>
        <member name="83832744787">
          <property name="role" value="passive"/>
        </member>
      </group>
    </meeting>
  </response>
</out>

```

```
</response>  
</out>
```

[Hc_tc_meetings_list](#)

Команда предназначена для получения списка совещаний.

Аналог команды в CoCon:

domain/<DOMAIN>/tc/meeting/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/tc_meetings_list

Код ответа:

200 - успешно;

404 - ошибка.

XSD-схема, валидирующая входные и выходные данные:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- Types -->
  <xs:complexType name="meetingType">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="meeting" type="meetingType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/tc_meetings_list

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="tc_meetings_list.xsd">
  <request domain="biysk.local"/>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_tc_meetings_list.xsd">
  <response>
    <meeting name="default" id="default" description=""/>
    <meeting name="test_tc2" id="83624A7E487" description="web_tc"/>
    <meeting name="big_meeting" id="836235ECD74" description="&quot;for task 85999&quot;"/>
    <meeting name="240101_tc1" id="836263BD418" description="для менеджера 240101"/>
    <meeting name="tc_240519" id="83622B5102B" description="test 240519 + 240475"/>
    <meeting name="tc_240475" id="8362572E53B" description="проверка на простом ТА"/>
    <meeting name="test_ewsd_tc" id="8362096EF92" description="проверка селектора для инженеров EWSD"/>
  </response>
</out>
```

Нс_tc_meeting - Управление шаблонами совещаний Teleconference на виртуальной АТС

- [Нс_tc_meeting](#)
- [Нс_tc_meeting_remove](#)

Нс_tc_meeting

Команда позволяет создавать/редактировать шаблон совещания на виртуальной АТС.

Аналог команды в CoCon:

domain/<DOMAIN>/tc/meeting/.meeting

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/tc_meeting

Код HTTP-ответа:

- 201 - успешно;
- 404 - ошибка.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="common_types.xsd"/>
  <xs:include schemaLocation="error.xsd"/>
  <!-- Types -->
  <xs:complexType name="memberType">
    <xs:sequence>
      <xs:element name="property" type="propertyType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="groupType">
    <xs:sequence>
      <xs:element name="member" type="memberType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:integer" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="masterType">
    <xs:attribute name="option" type="xs:string" use="required"/>
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="meetingType">
    <xs:sequence>
      <xs:element name="group" type="groupType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="master" type="masterType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="optional"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="greeting_url" type="xs:string" use="optional"/>
    <xs:attribute name="is_public" type="xs:boolean" use="required"/>
    <xs:attribute name="is_open" type="xs:boolean" use="required"/>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="meeting" type="meetingType"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:simpleType name="responseType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="tc meetings declare ok"/>
    </xs:restriction>
  </xs:simpleType>

```

```

        </xs:restriction>
    </xs:simpleType>

    <!-- in -->
    <xs:element name="in">
        <xs:complexType>
            <xs:all>
                <xs:element name="request" type="requestType"/>
            </xs:all>
        </xs:complexType>
    </xs:element>

    <!-- out -->
    <xs:element name="out">
        <xs:complexType>
            <xs:choice>
                <!--Positive case-->
                <xs:element name="response">
                    <xs:complexType>
                        <xs:all>
                            <xs:element name="response" type="responseType"/>
                        </xs:all>
                    </xs:complexType>
                </xs:element>
                <!--Or error-->
                <xs:element name="error" type="errorType"/>
            </xs:choice>
        </xs:complexType>
    </xs:element>
</xs:schema>

```

Пример

Запрос на создание нового совещания:

http://192.168.1.21:9999/commands/tc_meeting

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- new meeting -->
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_tc_me
eting.xsd">
  <request domain="biysk.local">
    <meeting name="new_meeting2" description="мое новое совещание" greeting_url="system://
sounds/ai_participant_of_conference.wav" is_public="true" is_open="true">
      <group id="1" name="Group One">
        <member name="240502">
          <property name="role" value="passive" />
        </member>
        <member name="240503">
          <property name="role" value="active" />
        </member>
        <member name="83832744787">
          <property name="role" value="passive" />
        </member>
      </group>
    </meeting>
  </request>
</in>
```

Ответ:201

Запрос редактирование совещания 83620F86F67:

http://192.168.1.21:9999/commands/tc_meeting

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- edit meeting -->
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="hc_tc_me
eting.xsd">
  <request domain="biysk.local">
    <meeting id="83620F86F67" name="meeting_2" description="мое отредактированное совещание"
greeting_url="system://sounds/ai_participant_of_conference.wav" is_public="true" is_open="false">
      <group id="2" name="Group Two">
        <member name="240505">
          <property name="role" value="passive" />
        </member>
        <member name="240506">
          <property name="role" value="active" />
        </member>
        <member name="89913699011">
          <property name="role" value="passive" />
        </member>
      </group>
    </meeting>
  </request>
</in>

```

Ответ:

В случае успеха будет 201 ответ, в случае ошибки - 404

hc_tc_meeting_remove

Команда позволяет удалять шаблон совещания на виртуальной АТС.

Аналог команды в CoCon:

domain/<DOMAIN>/tc/meeting/.remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/tc_meeting_remove

Код HTTP-ответа:

- 201 - успешно;
- 404 - ошибка.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:complexType name="meetingType">
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="errorType">
    <xs:attribute name="error" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="meeting" type="meetingType"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="responseType">
    <xs:attribute name="error" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Запрос:

http://192.168.1.21:9999/commands/tc_meeting_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="tc_meeting_remove.xsd">
  <request domain="biysk.local">
    <meeting id="83626ECEF82" />
  </request>
</in>
```

Ответ:

В случае успеха будет 201 ответ, в случае ошибки 404 + xml с описанием ошибки:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="hc_tc_meeting_remove.xsd">
  <response error="not_found"/>
</out>
```

hc_tc_members_list - Просмотр списков всех участников и ведущих в домене

tc_members_list

Команда предназначена для просмотра списков всех участников и ведущих в домене сервиса селекторной связи (Teleconference).

Аналог команды в CoCon:

`domain/<DOMAIN>/tc/member/list`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://server:9999/commands/tc_members_list`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- Types -->
  <xs:simpleType name="respMemberType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="member"/>
      <xs:enumeration value="master"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="reqMemberType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="members"/>
      <xs:enumeration value="masters"/>
      <xs:enumeration value="all"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="type" type="reqMemberType" use="optional"/>
  </xs:complexType>

  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="member" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="type" type="respMemberType" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

```



```

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:all>
      <xs:element name="response" type="responseType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/tc_members_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="tc_members_list.xsd">
  <request domain="biysk.local"/>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_tc_members_list.xsd">
  <response>
    <member name="240101" type="potential_master"/>
    <member name="240466" type="potential_master"/>
    <member name="240467" type="potential_master"/>
    <member name="240473" type="potential_master"/>
  </response>
</out>

```

Нс_tc_options - Управление настройками телеконференции уровня кластера ядра

- [Нс_tc_options_clean](#)
- [Нс_tc_options_info](#)
- [Нс_tc_options_set](#)

Нс_tc_options_clean

Команда сброса настроек телеконференции уровня кластера ядра.

Аналог команды в CoCon:

system/tc/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tc_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="ns_options" type="nsCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ns_options" type="nsCleanOutType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/tc_options_clean

```

<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tc_options_clean.xsd">
  <request>
    <ns_options>
      <system/>
      <node name="core1@Artem-Work-PC">
        <option name="count_active_channels"/>
        <option name="enabled"/>
        <option name="mysql_database"/>
        <option name="mysql_enabled"/>
        <option name="mysql_host"/>
        <option name="mysql_password"/>
        <option name="mysql_port"/>
        <option name="mysql_timeout"/>
        <option name="mysql_user"/>
      </node>
    </ns_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tc_options_clean.xsd">
  <result>
    <ns_options>
      <system/>
      <node name="core1@Artem-Work-PC">
        <success option="count_active_channels"/>
        <success option="enabled"/>
        <success option="mysql_database"/>
        <success option="mysql_enabled"/>
        <success option="mysql_host"/>
        <success option="mysql_password"/>
        <success option="mysql_port"/>
        <success option="mysql_timeout"/>
        <success option="mysql_user"/>
      </node>
    </ns_options>
  </result>
</out>

```

Нс_tc_options_info

Команда просмотра информации свойствах телеконференции уровня кластера ядра.

Аналог команды в CoCon:

system/tc/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tc_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- tcOptionsType -->
  <xs:complexType name="tcOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="enabled" type="booleanType"/>
        <xs:attribute name="count_active_channels" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              infinity | pos_integer()
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="mysql_enabled" type="booleanType"/>
        <xs:attribute name="mysql_host" type="stringType"/>
        <xs:attribute name="mysql_port" type="inetPortType"/>
        <xs:attribute name="mysql_user" type="stringType"/>
        <xs:attribute name="mysql_password" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              In case of strong security restrictions password may be shown as
              *****
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="mysql_database" type="stringType"/>
        <xs:attribute name="mysql_timeout" type="integerType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="ns_options" type="nsInfoInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```

</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="ns_options" type="nsInfoOutType"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/tc_options_info

```

<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tc_options_info.xsd">
  <request>
    <ns_options node="core1@ecss1"/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tc_options_info.xsd">
  <result>
    <ns_options>
      <system>
        <options xs:type="tcOptionsType"/>
      </system>
      <node name="core1@ecss1">
        <options xs:type="tcOptionsType" mysql_user="history_db" mysql_timeout="300000"
mysql_port="3306" mysql_password="*****" mysql_host="tc.mysql.ecss" mysql_enabled="true"
mysql_database="history_db" enabled="true" count_active_channels="infinity"/>
      </node>
    </ns_options>
  </result>
</out>

```

Нс_tc_options_set

Команда изменения настроек телеконференции уровня кластера ядра.

Аналог команды в CoCon:

system/tc/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tc_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- tcOptionsType -->
  <xs:complexType name="tcOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="enabled" type="booleanType"/>
        <xs:attribute name="count_active_channels" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              infinity | pos_integer()
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="mysql_enabled" type="booleanType"/>
        <xs:attribute name="mysql_host" type="stringType"/>
        <xs:attribute name="mysql_port" type="inetPortType"/>
        <xs:attribute name="mysql_user" type="stringType"/>
        <xs:attribute name="mysql_password" type="stringType"/>
        <xs:attribute name="mysql_database" type="stringType"/>
        <xs:attribute name="mysql_timeout" type="integerType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="ns_options" type="nsSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>

```

```

        <xs:sequence>
            <xs:element name="ns_options" type="nsSetOutType"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
    <xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/tc_options_set

```

<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tc_options_set.xsd">
    <request>
        <ns_options>
            <system>
                <options xs:type="tcOptionsType"/>
            </system>
            <node name="core1@Artem-Work-PC">
                <options count_active_channels="infinity" enabled="true"
mysql_database="history_db" mysql_enabled="false" mysql_host="tc.mysql.ecss"
mysql_password="history_db" mysql_port="3306" mysql_timeout="300000" mysql_user="history_db"
                xs:type="tcOptionsType"/>
            </node>
        </ns_options>
    </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tc_options_set.xsd">
  <result>
    <ns_options>
      <system/>
      <node name="core1@Artem-Work-PC">
        <success option="count_active_channels"/>
        <success option="enabled"/>
        <success option="mysql_database"/>
        <success option="mysql_enabled"/>
        <success option="mysql_host"/>
        <success option="mysql_password"/>
        <success option="mysql_port"/>
        <success option="mysql_timeout"/>
        <success option="mysql_user"/>
      </node>
    </ns_options>
  </result>
</out>

```

Нс_tc_phones - Управление телефонами сервиса "Селекторной связи"

- [tc_phone_push](#)
- [tc_phone_status](#)

[tc_phone_push](#)

Команда обновления конфигурации телефона.

Аналог команды в CoCon:

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/tc_phone_push

Код HTTP-ответа:

- reason = ok - успешно;
- reason = other - ошибка.

XSD-схема, валидирующая входные и выходные данные:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- Types -->
  <xs:complexType name="phonesStatusType">
    <xs:attribute name="phone" type="xs:string" use="required"/>
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="master" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="meeting" type="phonesStatusType" minOccurs="1"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Примеры:

Запрос на обновление конфигурации телефонов всех мастеров в домене:
http://192.168.1.21:9999/commands/tc_phone_push

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_tc_phones_push.xsd">
  <request domain="biysk.local"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_tc_phones_push.xsd">
  <response>
    <phone phone="240101" reason="ok"/>
    <phone phone="240244" reason="ok"/>
    <phone phone="240475" reason="ok"/>
  </response>
</out>
```

Запрос на обновление конфигурации телефона мастера 240101 в домене:

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_tc_phone_push.xsd">
  <request domain="biysk.local" master="240101"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_tc_phones_push.xsd">
  <response>
    <phone phone="240101" reason="ok"/>
  </response>
</out>
```

tc_phone_status

Команда просмотра статуса телефона.

Аналог команды в CoCon:

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://server:9999/commands/tc_phone_status

Код HTTP-ответа:

- 200 - успешно;
- 404 - ошибка.

XSD-схема, валидирующая входные и выходные данные:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- Types -->
  <xs:complexType name="phonesStatusType">
    <xs:attribute name="phone" type="xs:string" use="required"/>
    <xs:attribute name="alias" type="xs:string" use="required"/>
    <xs:attribute name="api" type="xs:string" use="required"/>
    <xs:attribute name="uac" type="xs:string" use="required"/>
    <xs:attribute name="ip" type="xs:string" use="optional"/>
    <xs:attribute name="status" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="responseType">
    <xs:sequence>
      <xs:element name="meeting" type="phonesStatusType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="responseType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос статусов телефонов мастеров из домена: http://192.168.1.21:9999/commands/tc_phone_status

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_tc_phone_status_list.xsd">
  <request domain="biysk.local"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_tc_phones_status_list.xsd">
  <response>
    <phone phone="240101" alias="fbbaa0f2252e4139" api="tc_yealink_api" uac="Yealink SIP-T28P
2.73.14.3" ip="192.168.2.6" status="configured"/>
    <phone phone="240244" alias="fbba9fd42d8ffffb5" api="tc_yealink_api" uac="Yealink SIP-T28P
2.73.14.3" ip="192.168.2.6" status="configured"/>
    <phone phone="240475" alias="fbbaa155aa3145fb" api="tc_undefined_api" uac="RG-1402G/1.8.1
SN/VI0C000069 sofia-sip/1.12.10" ip="192.168.2.6" status="configured"/>
  </response>
</out>
```

Нс_tc_system_options - Управление настройками Телеконференции на уровне системы

- [Нс_tc_system_options_clean](#)
- [Нс_tc_system_options_info](#)
- [Нс_tc_system_options_set](#)

Нс_tc_system_options_clean

Команда сброса системных настроек телеконференции.

Аналог команды в CoCon:

system/tc/properties/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tc_system_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean system tc options.
      Usage: http://server:9999/commands/tc_system_options_clean.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- tcSystemOptionsType -->
  <xs:complexType name="tcSystemOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="rest_host" type="stringType"/>
        <xs:attribute name="rest_port" type="inetPortType"/>
        <xs:attribute name="session_lifetime" type="integerType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="ns_options" type="nsCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ns_options" type="nsCleanOutType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/tc_system_options_clean

```

<?xml version="1.0"?>
  <in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tc_system_options_clean.xsd">
    <request>
      <ns_options>
        <system/>
        <node name="core1@Artem-Work-PC">
          <option name="session_lifetime"/>
          <option name="rest_port"/>
          <option name="rest_host"/>
        </node>
      </ns_options>
    </request>
  </in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
  <out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tc_system_options_clean.xsd">
    <result>
      <ns_options>
        <system/>
        <node name="core1@Artem-Work-PC">
          <success option="rest_host"/>
          <success option="rest_port"/>
          <success option="session_lifetime"/>
        </node>
      </ns_options>
    </result>
  </out>

```

Нс_tc_system_options_info

Команда просмотра системных настроек телеконференции.

Аналог команды в CoCon:

system/tc/properties/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tc_system_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show system tc options.
      Usage: http://server:9999/commands/tc\_system\_options\_info.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- tcSystemOptionsType -->
  <xs:complexType name="tcSystemOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="rest_host" type="stringType"/>
        <xs:attribute name="rest_port" type="inetPortType"/>
        <xs:attribute name="session_lifetime" type="integerType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="ns_options" type="nsInfoInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ns_options" type="nsInfoOutType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```

    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/tc_system_options_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="sorm_system_options_info.xsd">
  <request>
    <system_options/>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tc_system_options_info.xsd">
  <result>
    <ns_options>
      <system>
        <options xs:type="tcSystemOptionsType"/>
      </system>
      <node name="core1@Artem-Work-PC">
        <options xs:type="tcSystemOptionsType" session_lifetime="1800" rest_port="9999"
rest_host="ecss1"/>
      </node>
    </ns_options>
  </result>

```

[Hc_tc_system_options_set](#)

Команда изменения системных настроек телеконференции.

Аналог команды в CoCon:

system/tc/properties/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tc_system_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set system tc options.
      Usage: http://server:9999/commands/tc_system_options_set.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>

  <!-- tcSystemOptionsType -->
  <xs:complexType name="tcSystemOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="rest_host" type="stringType"/>
        <xs:attribute name="rest_port" type="inetPortType"/>
        <xs:attribute name="session_lifetime" type="integerType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="ns_options" type="nsSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ns_options" type="nsSetOutType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```



```

    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/tc_system_options_set

```

<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tc_system_options_set.xsd">
  <request>
    <ns_options>
      <system>
        <options xs:type="tcSystemOptionsType" />
      </system>
      <node name="core1@Artem-Work-PC">
        <options xs:type="tcSystemOptionsType" session_lifetime="1800" rest_port="9999"
rest_host="ecss1"/>
      </node>
    </ns_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```

<?xml version="1.0"?>
  <out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tc_system_options_set.xsd">
    <result>
      <ns_options>
        <system/>
        <node name="core1@Artem-Work-PC">
          <success option="rest_host"/>
          <success option="rest_port"/>
          <success option="session_lifetime"/>
        </node>
      </ns_options>
    </result>
  </out>

```

Нс_tc_web_options - Управление веб-свойствами телеконференции

- [tc_web_options_clean](#)
- [tc_web_options_info](#)
- [tc_web_options_set](#)

[tc_web_options_clean](#)

Очистить веб свойства телеконференции.

Аналог команды в CoSop:

domain/<DOMAIN>/tc/properties/web/clean

Метод HTTP-запроса:

POST

Шаблон http запроса

http://host:port/commands/tc_web_options_clean

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean tc web options.
      Usage: http://server:9999/commands/tc\_web\_options\_clean.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:all>
      <xs:element name="domain_options" type="domainCleanOutType" minOccurs="1"/>
    </xs:all>
  </xs:complexType>
  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" type="resultType"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/tc_web_options_clean

```

<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="tc_web_options_clean.xsd">
  <request>
    <domain_options>
      <system />
      <domain name="biysk.local">
        <option name="notification_timeout"/>
        <option name="member_adittional_fields"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="tc_web_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="member_adittional_fields"/>
        <success option="notification_timeout"/>
      </domain>
    </domain_options>
  </result>
</out>

```

tc_web_options_info

Показать веб свойства телеконференции.

Аналог команды в CoCon:

domain/<DOMAIN>/tc/properties/web/info

Метод HTTP-запроса:

POST

Шаблон http запроса

http://host:port/commands/tc_web_options_info

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show tc web options.
      Usage: http://server:9999/commands/tc\_web\_options\_info.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- tcWebOptionsType -->
  <xs:complexType name="tcWebOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="member_adittional_fields" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              usage: field1;field2;field3;
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="notification_timeout" type="integerType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainInfoInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>

```

```

        <xs:all>
            <xs:element name="domain_options" type="domainInfoOutType"/>
        </xs:all>
    </xs:complexType>
</xs:element>
    <xs:element name="error" type="errorType" />
</xs:choice>
</xs:complexType>
</xs:element>
<!-- out -->
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/tc_web_options_info

```

<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="tc_web_options_info.xsd">
    <request>
        <domain_options domain="biysk.local"/>
    </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="tc_web_options_info.xsd">
    <result>
        <domain_options>
            <system>
                <options xs:type="tcWebOptionsType" notification_timeout="10"
member_adittional_fields="departament_code;departament_name;position"/>
            </system>
        </domain_options>
    </result>
</out>

```

tc_web_options_set

Изменить веб свойства телеконференции.

Аналог команды в CoCon:

domain/<DOMAIN>/tc/properties/web/set

Метод HTTP-запроса:

POST

Шаблон http запроса

http://host:port/commands/tc_web_options_set

Код http ответа

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема XML запроса/ответа


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set tc web options.
      Usage: http://server:9999/commands/tc\_web\_options\_set.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- tcWebOptionsType -->
  <xs:complexType name="tcWebOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="member_adittional_fields" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              usage: field1;field2;field3;
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="notification_timeout" type="integerType"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>

```

```

        <xs:all>
            <xs:element name="domain_options" type="domainSetOutType"/>
        </xs:all>
    </xs:complexType>
</xs:element>
    <xs:element name="error" type="errorType" />
</xs:choice>
</xs:complexType>
</xs:element>
<!-- out -->
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/tc_web_options_set

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="tc_web_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="tcWebOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options notification_timeout="15" member_adittional_fields="departament_code;departament_name;position" xs:type="tcWebOptionsType"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="tc_web_opt
ions_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="member_adittional_fields"/>
        <success option="notification_timeout"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Hc_tracer - Управление Tracer

- [Hc_tracer_declare](#)
- [Hc_tracer_show](#)
- [Hc_tracer_remove](#)

Hc_tracer_declare

Команда позволяет создать трейсер вызова в системе.

Аналог команды в CoCon:

```

cluster/core/<CORE>/tracer/declare
domain/<DOMAIN>/tracer/declare

```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tracer_declare

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

trace_common.xsd

Hc_tracer_show

tracer_declare.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="tracer_common.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="tracer" type="tracerType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="ok">
          <xs:complexType/>
        </xs:element>
        <xs:element name="already_defined">
          <xs:complexType/>
        </xs:element>
        <xs:element name="bad_domain">
          <xs:complexType/>
        </xs:element>
        <xs:element name="bad_tracer">
          <xs:complexType/>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

В случае успешного ответа возвращается код 200 и ответ </ok>.

Если трейсер с таким именем уже объявлен - возвращается код 200 и ответ </already_defined>.
 Если во время создания трейса произошла ошибка - возвращается код 200 и ответ </bad_tracer>.

Пример:

Запрос:

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="tracer_declare.xsd">
  <request>
    <tracer addr="240465" domain="biysk.local" node="core1@ecss1" name="tr1" lifetime="1"
  iface="" >
      <writers>
        <writer xsi:type="fileTracerType" path="1.txt" write_type="write"/>
      </writers>
    </tracer>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tracer_declare.xsd">
  <ok/>
</out>
```

Повторный запрос возвращает код 200, и ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tracer_declare.xsd">
  <already_defined/>
</out>
```

Ис_tracer_show

Команда позволяет посмотреть список tracer-ов, активных в системе.

Аналог команды в CoCon:

```
cluster/core/<CORE>/tracer/show
domain/<DOMAIN>/tracer/show
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tracer_show

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

trace_show.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- writeTypeType -->
  <xs:simpleType name="writeTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="append"/>
      <xs:enumeration value="write"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- baseTracerType-->
  <xs:complexType name="baseTracerType" abstract="true"/>
  <!-- udpTracerType-->
  <xs:complexType name="udpTracerType">
    <xs:complexContent>
      <xs:extension base="baseTracerType">
        <xs:attribute name="host" type="xs:string" use="required"/>
        <xs:attribute name="port" type="xs:integer" use="required"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- fileTracerType-->
  <xs:complexType name="fileTracerType">
    <xs:complexContent>
      <xs:extension base="baseTracerType">
        <xs:attribute name="path" type="xs:string" use="required"/>
        <xs:attribute name="write_type" type="writeTypeType" use="required"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- mysqlTracerType-->
  <xs:complexType name="mysqlTracerType">
    <xs:complexContent>
      <xs:extension base="baseTracerType">
        <xs:attribute name="db" type="xs:string" use="required"/>
        <xs:attribute name="user" type="xs:string" use="required"/>
        <xs:attribute name="passwd" type="xs:string" use="required"/>
        <xs:attribute name="host" type="xs:string" use="required"/>
        <xs:attribute name="port" type="xs:integer" use="required"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <!-- tracerType-->
  <xs:complexType name="tracerType">
    <xs:sequence>
      <xs:element name="writers">
        <xs:complexType>
          <xs:sequence>

```

```
        <xs:element name="writer" type="baseTracerType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="created" minOccurs="0" maxOccurs="1">
    <xs:complexType>
        <xs:attribute name="second" type="xs:integer" use="required"/>
        <xs:attribute name="minute" type="xs:integer" use="required"/>
        <xs:attribute name="hour" type="xs:integer" use="required"/>
        <xs:attribute name="day" type="xs:integer" use="required"/>
        <xs:attribute name="month" type="xs:integer" use="required"/>
        <xs:attribute name="year" type="xs:integer" use="required"/>
    </xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="domain" type="xs:string" use="required"/>
<xs:attribute name="name" type="xs:string" use="required"/>
<xs:attribute name="node" type="xs:string" use="required"/>
<xs:attribute name="iface" type="xs:string" use="optional"/>
<xs:attribute name="addr" type="xs:string" use="optional"/>
<xs:attribute name="lifetime" type="xs:integer" use="optional"/>
</xs:complexType>
</xs:schema>
```

tracer_common.xsd


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="tracer_common.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="tracers">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="tracer" type="tracerType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/tracer_show

```

<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="tracer_show.xsd">
  <request domain="biysk.local"/>
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tracer_show.xsd">
  <tracers>
    <tracer name="tr1" domain="biysk.local" node="core1@ecss1" lifetime="1" iface=""
  addr="240465">
      <writers>
        <writer xs:type="fileTracerType" path="1.txt" write_type="write"/>
      </writers>
      <created second="43" minute="52" hour="7" day="28" month="6" year="2019"/>
    </tracer>
  </tracers>
</out>
```

hc_tracer_remove

Команда позволяет удалить из системы tracer по имени.

Аналог команды в CoCon:

```
cluster/core/<CORE>/tracer/remove
domain/<DOMAIN>/tracer/remove
```

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tracer_remove

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="tracer_common.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="tracer" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос:

http://192.168.1.21:9999/commands/tracer_remove

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="tracer_remove.xsd">
  <request domain="biysk.local" tracer="tr1"/>
</in>
```

Код ответа:

204

Hc_trunk - Команды управления транками

- [Hc_trunk_clean](#)
- [Hc_trunk_info](#)
- [Hc_trunk_set](#)
- [Hc_trunk_list](#)

Hc_trunk_clean

Команда позволяет для определенного свойства транка установить значение по умолчанию.

Аналог команды в CoCon:

domain/<DOMAIN>/trunk/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/trunk_clean

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XML-схема:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="property" type="trunkProperty" minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="trunk" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="trunkProperty">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="success" type="xs:string"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Примеры

Запрос:

http://192.168.1.21:9999/commands/trunk_set

```
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_trunk_clean.xsd">
  <request domain="biysk.local" trunk="bsk1">
    <property name="black_list\out\enabled"/>
    <property name="CPS_limit"/>
  </request>
</in>
```

Ответ:200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_trunk_clean.xsd">
  <success/>
</out>
```

hc_trunk_info

Команда позволяет просматривать конфигурации определенного транка и информацию о транке в реальном времени.

Аналог команды в CoSop:

domain/<DOMAIN>/trunk/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/trunk_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XML-схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="requestType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="trunk" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="trunkProperty">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="trunkOutType">
    <xs:sequence>
      <xs:element name="property" type="trunkProperty" minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="group" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:all>
              <xs:element name="trunk" type="trunkOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Примеры

Запрос:

http://192.168.1.21:9999/commands/trunk_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_trunk_info.xsd">
  <request domain="biysk.local" trunk="smg-4"/>
</in>
```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_trunk_info.xsd">
  <response>
    <trunk name="smg-4" group="smg.gr">
      <property name="system\bandwidth\in" value="unbounded"/>
      <property name="system\bandwidth\out" value="unbounded"/>
      <property name="system\bandwidth\total" value="unbounded"/>
      <property name="bandwidth\in" value="unbounded"/>
      <property name="bandwidth\out" value="unbounded"/>
      <property name="bandwidth\total" value="256"/>
      <property name="actual\bandwidth\in" value="unbounded"/>
      <property name="actual\bandwidth\out" value="unbounded"/>
      <property name="actual\bandwidth\total" value="256"/>
      <property name="cps_limit\in" value="256"/>
      <property name="cps_limit\out" value="256"/>
      <property name="cps_limit\total" value="256"/>
      <property name="type" value="sip"/>
      <property name="owner" value="sip1"/>
      <property name="black_list\in\enabled" value="false"/>
      <property name="black_list\out\enabled" value="false"/>
      <property name="white_list\in\enabled" value="false"/>
      <property name="white_list\out\enabled" value="false"/>
      <property name="in" value="0"/>
      <property name="out" value="0"/>
      <property name="cps\in" value="0"/>
      <property name="cps\out" value="0"/>
      <property name="stat\in" value="4911"/>
      <property name="stat\out" value="34"/>
      <property name="stat\cps_max\in" value="2"/>
      <property name="stat\cps_max\out" value="1"/>
      <property name="stat\cps_max\total" value="2"/>
      <property name="stat\rejected\in" value="0"/>
      <property name="stat\rejected\out" value="0"/>
    </trunk>
  </response>
</out>

```

hc_trunk_set

Команда позволяет выполнить настройку конфигурации определенного транка.

Аналог команды в CoCon:

domain/<DOMAIN>/trunk/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/trunk_set

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XML-схема:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>
  <xs:complexType name="requestType">
    <xs:sequence>
      <xs:element name="property" type="trunkProperty" minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="trunkProperty">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="trunk" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="trunkProperty">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="requestType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="success" type="xs:string"/>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Примеры

Запрос:

http://192.168.1.21:9999/commands/trunk_set

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_trunk_set.xsd">
  <request domain="biysk.local" trunk="bsk1">
    <property name="black_list\out\enabled" value="true"/>
    <property name="CPS_limit" value="10"/>
  </request>
</in>
```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_trunk_set.xsd">
  <success/>
</out>
```

hc_trunk_list

Команда возвращает список транковых интерфейсов, определенных для конкретного домена.

Аналог команды в CoCon:

`/domain/<DOMAIN>/trunk/list`

Метод http запроса:

POST

Шаблон http запроса:

`http://host:port/commands/trunk_list`

Код http ответа:

200 – в случае успеха;

404 – в случае ошибки.

Xsd схема xml файла с запросом/ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- ifaceType -->
  <xs:complexType name="ifaceType">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="owner" type="xs:string" use="required"/>
    <xs:attribute name="group" type="xs:string" use="required"/>
    <xs:attribute name="type" type="xs:string" use="required"/>
    <xs:attribute name="active" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- ifacesType -->
  <xs:complexType name="ifacesType">
    <xs:sequence>
      <xs:element name="iface" type="ifaceType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="owner" type="xs:string" use="optional"/>
            <xs:attribute name="group" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="ifaces" type="ifacesType"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример запроса всех транков в определенном домене:

Запрос: http://192.168.1.21:9999/commands/trunk_list

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_trunk_list.xsd">
  <request domain="biysk.local"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_trunk_list.xsd">
  <ifaces>
    <iface id="smg-4" name="smg-4" owner="sip1" group="smg.gr" type="sip" active="true"/>
    <iface id="to_aster" name="to_aster" owner="sip1" group="sbc.gr" type="sip" active="true"
  />
    <iface id="ems1" name="ems1" owner="sip1" group="ssw.gr" type="sip" active="true"/>
    <iface id="ems2" name="ems2" owner="sip1" group="ssw.gr" type="sip" active="true"/>
    <iface id="bsk1" name="bsk1" owner="sip1" group="ssw.gr" type="sip" active="true"/>
    <iface id="to_sipp" name="to_sipp" owner="sip1" group="sbc.gr" type="sip" active="true"/>
    <iface id="bsk2" name="bsk2" owner="sip1" group="ssw.gr" type="sip" active="true"/>
    <iface id="nsk_sbc" name="nsk_sbc" owner="sip1" group="sbc.gr" type="sip" active="true"/>
    <iface id="tr_speaker" name="tr_speaker" owner="sip1" group="smg.gr" type="sip"
  active="true"/>
  </ifaces>
</out>
```

Пример запроса всех транков по определенному владельцу в определенном домене:

Запрос: http://192.168.1.21:9999/commands/trunk_list

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_trunk_list.xsd">
  <request domain="biysk.local" owner="sip1"/>
</in>
```

Ответ:200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_trunk_list.xsd">
  <ifaces>
    <iface id="smg-4" name="smg-4" owner="sip1" group="smg.gr" type="sip" active="true"/>
    <iface id="to_aster" name="to_aster" owner="sip1" group="sbc.gr" type="sip" active="true"
  />

    <iface id="ems1" name="ems1" owner="sip1" group="ssw.gr" type="sip" active="true"/>
    <iface id="ems2" name="ems2" owner="sip1" group="ssw.gr" type="sip" active="true"/>
    <iface id="bsk1" name="bsk1" owner="sip1" group="ssw.gr" type="sip" active="true"/>
    <iface id="to_sipp" name="to_sipp" owner="sip1" group="sbc.gr" type="sip" active="true"/>
    <iface id="bsk2" name="bsk2" owner="sip1" group="ssw.gr" type="sip" active="true"/>
    <iface id="nsk_sbc" name="nsk_sbc" owner="sip1" group="sbc.gr" type="sip" active="true"/>
    <iface id="tr_speaker" name="tr_speaker" owner="sip1" group="smg.gr" type="sip"
  active="true"/>
  </ifaces>
</out>

```

Пример запроса всех транков по определенной группе и домену:

Запрос: http://192.168.1.21:9999/commands/trunk_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_trunk_list.xsd">
  <request domain="biysk.local" group="sbc.gr"/>
</in>

```

Ответ:200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_trunk_list.xsd">
  <ifaces>
    <iface id="to_aster" name="to_aster" owner="sip1" group="sbc.gr" type="sip" active="true"
  />

    <iface id="to_sipp" name="to_sipp" owner="sip1" group="sbc.gr" type="sip" active="true"/>
    <iface id="nsk_sbc" name="nsk_sbc" owner="sip1" group="sbc.gr" type="sip" active="true"/>
  </ifaces>
</out>

```

Нс_tts_cluster_options - Управление опцией подсистемы TTS уровня кластера

- [Hc_tts_cluster_options_info](#)
- [Hc_tts_cluster_options_set](#)
- [Hc_tts_cluster_options_clean](#)

[Hc_tts_cluster_options_info](#)

Просмотр списка опций подсистемы TTS уровня кластера.

Аналог команды в CoCon:

`cluster/core/<CLUSTER>/tts/db/info`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/tts_cluster_options_info`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="tts_cluster_options_info">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Get options of TTS node.
      Usage: http://server:9999/commands/tts_cluster_options_info
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <!-- Options for TTS node -->
  <xs:complexType name="TtsClusterOptions">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="mysql_database" type="nonEmptyNormalizedStringType"
use="optional" />
        <xs:attribute name="mysql_enabled" type="booleanType" use="optional" />
        <xs:attribute name="mysql_host" type="nonEmptyNormalizedStringType"
use="optional" />
        <xs:attribute name="mysql_password" type="stringType">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              In case of strong security restrictions password may be shown as
*****
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="mysql_port" type="inetPortType" use="optional" />
        <xs:attribute name="mysql_user" type="nonEmptyNormalizedStringType"
use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterInfoInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

```
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:all minOccurs="1" maxOccurs="1">
            <xs:element name="cluster_options" type="clusterInfoOutType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>
```

Пример:

Запрос:

http://192.168.1.21:9999/commands/tts_cluster_options_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in>
  <request>
    <cluster_options cluster="core1"/>
  </request>
</in>
```

Код ответа:

200

Ответ:


```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_tts_cluster_options_info.xsd">
  <result>
    <cluster_options>
      <cluster name="core1">
        <options xs:type="TtsClusterOptions" mysql_database="ecss_calls_db"
mysql_enabled="true" mysql_host="tts.mysql.ecss" mysql_password="*****" mysql_port="3306"
mysql_user="calls_db"/>
      </cluster>
    </cluster_options>
  </result>
</out>
```

hc_tts_cluster_options_set

Установка значения опций подсистемы TTS уровня кластера.

Аналог команды в CoCon:

cluster/core/<CLUSTER>/tts/db/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tts_cluster_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- * 404 – в случае ошибки.

XSD-схема XML-запроса/ответа hc_tts_cluster_options_set.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="tts_cluster_options_set">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Set tts cluster options.
      Usage: http://server:9999/commands/tts\_cluster\_options\_set
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <xs:complexType name="TtsClusterOptions">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="mysql_database" type="nonEmptyNormalizedStringType"
use="optional" />
        <xs:attribute name="mysql_enabled" type="booleanType" use="optional" />
        <xs:attribute name="mysql_host" type="nonEmptyNormalizedStringType"
use="optional" />
        <xs:attribute name="mysql_password" type="nonEmptyNormalizedStringType"
use="optional" />
        <xs:attribute name="mysql_port" type="inetPortType" use="optional" />
        <xs:attribute name="mysql_user" type="nonEmptyNormalizedStringType"
use="optional" />
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterSetInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">

```

```

        <xs:complexType>
          <xs:all minOccurs="1" maxOccurs="1">
            <xs:element name="cluster_options" type="clusterSetOutType"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/tts_cluster_options_set

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="hc_tts_cluster_options_set.xsd">
  <request>
    <cluster_options>
      <cluster name="core1">
        <options xs:type="TtsClusterOptions" mysql_host="tts.mysql.ecss"
mysql_password="calls_db" mysql_port="13306" mysql_user="calls_db"/>
      </cluster>
    </cluster_options>
  </request>
</in>

```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_tts_cluster_options_set.xsd">
  <result>
    <cluster_options>
      <cluster name="core1">
        <success option="mysql_user"/>
        <success option="mysql_port"/>
        <success option="mysql_password"/>
        <success option="mysql_host"/>
      </cluster>
    </cluster_options>
  </result>
</out>
```

hc_tts_cluster_options_clean

Сброс значения опций подсистемы TTS уровня кластера.

Аналог команды в CoCon:

cluster/core/<CLUSTER>/tts/db/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tts_cluster_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- * 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="tts_cluster_options_clean">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Clean tts cluster options.
      Usage: http://server:9999/commands/tts_cluster_options_clean
      Copyright (c) 2014, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterCleanInType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="cluster_options" type="clusterCleanOutType"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/tts_cluster_options_clean

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_tts_cluster_options_clean.xsd">
  <request>
    <cluster_options>
      <cluster name="core1">
        <option name="mysql_port"/>
      </cluster>
    </cluster_options>
  </request>
</in>
```

Код ответа:

200

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_tts_cluster_options_clean.xsd">
  <result>
    <cluster_options>
      <cluster name="core1">
        <success option="mysql_port"/>
      </cluster>
    </cluster_options>
  </result>
</out>
```

hc_tts_domain_options - Управление опцией подсистемы TTS уровня домена

- [hc_tts_domain_options_info](#)
- [hc_tts_domain_options_set](#)
- [hc_tts_domain_options_clean](#)

hc_tts_domain_options_info

Просмотр значений опций TTS уровня домена.

Аналог команды в CoCon:

/domain/<DOMAIN>/tts/properties/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tts_domain_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа `tts_domain_options_info.xsd`:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="tts_domain_options_info">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show TTS options.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <!--cdr_typeType-->
  <xs:simpleType name = "cdr_typeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="extended"/>
      <xs:enumeration value="normal"/>
      <xs:enumeration value="manual"/>
    </xs:restriction>
  </xs:simpleType>

  <!--port_sourceType-->
  <xs:simpleType name = "port_sourceType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="iface_name"/>
      <xs:enumeration value="iface_id"/>
    </xs:restriction>
  </xs:simpleType>

  <!--cdr_time_roundingType-->
  <xs:simpleType name = "cdr_time_roundingType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="downside"/>
      <xs:enumeration value="upside"/>
      <xs:enumeration value="downside_with_delta"/>
      <xs:enumeration value="upside_with_delta"/>
      <xs:enumeration value="mathematically"/>
      <xs:enumeration value="millisecond"/>
    </xs:restriction>
  </xs:simpleType>

  <!--cdr_field_delimiterType-->
  <xs:simpleType name = "cdr_field_delimiterType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="tab"/>
      <xs:enumeration value="space"/>
      <xs:enumeration value="comma"/>
      <xs:enumeration value="semicolon"/>
      <xs:enumeration value="8"/> <!-- ^H -->
    </xs:restriction>
  </xs:simpleType>

```



```
<xs:enumeration value="9"/> <!-- \t -->
<xs:enumeration value="10"/> <!-- \n -->
<xs:enumeration value="11"/> <!-- ^K -->
<xs:enumeration value="12"/> <!-- ^L -->
<xs:enumeration value="32"/> <!-- -->
<xs:enumeration value="33"/> <!-- ! -->
<xs:enumeration value="34"/> <!-- " -->
<xs:enumeration value="35"/> <!-- # -->
<xs:enumeration value="36"/> <!-- $ -->
<xs:enumeration value="37"/> <!-- % -->
<xs:enumeration value="38"/> <!-- & -->
<xs:enumeration value="39"/> <!-- ' -->
<xs:enumeration value="40"/> <!-- ( -->
<xs:enumeration value="41"/> <!-- ) -->
<xs:enumeration value="42"/> <!-- * -->
<xs:enumeration value="43"/> <!-- + -->
<xs:enumeration value="44"/> <!-- , -->
<xs:enumeration value="45"/> <!-- - -->
<xs:enumeration value="46"/> <!-- . -->
<xs:enumeration value="47"/> <!-- / -->
<xs:enumeration value="48"/> <!-- 0 -->
<xs:enumeration value="49"/> <!-- 1 -->
<xs:enumeration value="50"/> <!-- 2 -->
<xs:enumeration value="51"/> <!-- 3 -->
<xs:enumeration value="52"/> <!-- 4 -->
<xs:enumeration value="53"/> <!-- 5 -->
<xs:enumeration value="54"/> <!-- 6 -->
<xs:enumeration value="55"/> <!-- 7 -->
<xs:enumeration value="56"/> <!-- 8 -->
<xs:enumeration value="57"/> <!-- 9 -->
<xs:enumeration value="58"/> <!-- : -->
<xs:enumeration value="59"/> <!-- ; -->
<xs:enumeration value="60"/> <!-- < -->
<xs:enumeration value="61"/> <!-- = -->
<xs:enumeration value="62"/> <!-- > -->
<xs:enumeration value="63"/> <!-- ? -->
<xs:enumeration value="64"/> <!-- @ -->
<xs:enumeration value="65"/> <!-- A -->
<xs:enumeration value="66"/> <!-- B -->
<xs:enumeration value="67"/> <!-- C -->
<xs:enumeration value="68"/> <!-- D -->
<xs:enumeration value="69"/> <!-- E -->
<xs:enumeration value="70"/> <!-- F -->
<xs:enumeration value="71"/> <!-- G -->
<xs:enumeration value="72"/> <!-- H -->
<xs:enumeration value="73"/> <!-- I -->
<xs:enumeration value="74"/> <!-- J -->
<xs:enumeration value="75"/> <!-- K -->
<xs:enumeration value="76"/> <!-- L -->
<xs:enumeration value="77"/> <!-- M -->
```

```
<xs:enumeration value="78"/> <!-- N -->
<xs:enumeration value="79"/> <!-- O -->
<xs:enumeration value="80"/> <!-- P -->
<xs:enumeration value="81"/> <!-- Q -->
<xs:enumeration value="82"/> <!-- R -->
<xs:enumeration value="83"/> <!-- S -->
<xs:enumeration value="84"/> <!-- T -->
<xs:enumeration value="85"/> <!-- U -->
<xs:enumeration value="86"/> <!-- V -->
<xs:enumeration value="87"/> <!-- W -->
<xs:enumeration value="88"/> <!-- X -->
<xs:enumeration value="89"/> <!-- Y -->
<xs:enumeration value="90"/> <!-- Z -->
<xs:enumeration value="91"/> <!-- [ -->
<xs:enumeration value="92"/> <!-- \ -->
<xs:enumeration value="93"/> <!-- ] -->
<xs:enumeration value="94"/> <!-- ^ -->
<xs:enumeration value="95"/> <!-- _ -->
<xs:enumeration value="96"/> <!-- ` -->
<xs:enumeration value="97"/> <!-- a -->
<xs:enumeration value="98"/> <!-- b -->
<xs:enumeration value="99"/> <!-- c -->
<xs:enumeration value="100"/> <!-- d -->
<xs:enumeration value="101"/> <!-- e -->
<xs:enumeration value="102"/> <!-- f -->
<xs:enumeration value="103"/> <!-- g -->
<xs:enumeration value="104"/> <!-- h -->
<xs:enumeration value="105"/> <!-- i -->
<xs:enumeration value="106"/> <!-- j -->
<xs:enumeration value="107"/> <!-- k -->
<xs:enumeration value="108"/> <!-- l -->
<xs:enumeration value="109"/> <!-- m -->
<xs:enumeration value="110"/> <!-- n -->
<xs:enumeration value="111"/> <!-- o -->
<xs:enumeration value="112"/> <!-- p -->
<xs:enumeration value="113"/> <!-- q -->
<xs:enumeration value="114"/> <!-- r -->
<xs:enumeration value="115"/> <!-- s -->
<xs:enumeration value="116"/> <!-- t -->
<xs:enumeration value="117"/> <!-- u -->
<xs:enumeration value="118"/> <!-- v -->
<xs:enumeration value="119"/> <!-- w -->
<xs:enumeration value="120"/> <!-- x -->
<xs:enumeration value="121"/> <!-- y -->
<xs:enumeration value="122"/> <!-- z -->
<xs:enumeration value="123"/> <!-- { -->
<xs:enumeration value="124"/> <!-- | -->
<xs:enumeration value="125"/> <!-- } -->
<xs:enumeration value="126"/> <!-- ~ -->
<xs:enumeration value="160"/> <!-- -->
```

```

<xs:enumeration value="161"/> <!-- ¡ -->
<xs:enumeration value="162"/> <!-- ¢ -->
<xs:enumeration value="163"/> <!-- £ -->
<xs:enumeration value="164"/> <!-- ¤ -->
<xs:enumeration value="165"/> <!-- ¥ -->
<xs:enumeration value="166"/> <!-- ¦ -->
<xs:enumeration value="167"/> <!-- § -->
<xs:enumeration value="168"/> <!-- ¨ -->
<xs:enumeration value="169"/> <!-- © -->
<xs:enumeration value="170"/> <!-- ª -->
<xs:enumeration value="171"/> <!-- « -->
<xs:enumeration value="172"/> <!-- ¬ -->
<xs:enumeration value="173"/> <!-- -->
<xs:enumeration value="174"/> <!-- ® -->
<xs:enumeration value="175"/> <!-- ¯ -->
<xs:enumeration value="176"/> <!-- ° -->
<xs:enumeration value="177"/> <!-- ± -->
<xs:enumeration value="178"/> <!-- ² -->
<xs:enumeration value="179"/> <!-- ³ -->
<xs:enumeration value="180"/> <!-- ´ -->
<xs:enumeration value="181"/> <!-- µ -->
<xs:enumeration value="182"/> <!-- ¶ -->
<xs:enumeration value="183"/> <!-- · -->
<xs:enumeration value="184"/> <!-- , -->
<xs:enumeration value="185"/> <!-- ¹ -->
<xs:enumeration value="186"/> <!-- º -->
<xs:enumeration value="187"/> <!-- » -->
<xs:enumeration value="188"/> <!-- ¼ -->
<xs:enumeration value="189"/> <!-- ½ -->
<xs:enumeration value="190"/> <!-- ¾ -->
<xs:enumeration value="191"/> <!-- ¿ -->
<xs:enumeration value="192"/> <!-- À -->
<xs:enumeration value="193"/> <!-- Á -->
<xs:enumeration value="194"/> <!-- Â -->
<xs:enumeration value="195"/> <!-- Ã -->
<xs:enumeration value="196"/> <!-- Ä -->
<xs:enumeration value="197"/> <!-- Å -->
<xs:enumeration value="198"/> <!-- Æ -->
<xs:enumeration value="199"/> <!-- Ç -->
<xs:enumeration value="200"/> <!-- È -->
<xs:enumeration value="201"/> <!-- É -->
<xs:enumeration value="202"/> <!-- Ê -->
<xs:enumeration value="203"/> <!-- Ë -->
<xs:enumeration value="204"/> <!-- Ì -->
<xs:enumeration value="205"/> <!-- Í -->
<xs:enumeration value="206"/> <!-- Î -->
<xs:enumeration value="207"/> <!-- Ï -->
<xs:enumeration value="208"/> <!-- Ð -->
<xs:enumeration value="209"/> <!-- Ñ -->
<xs:enumeration value="210"/> <!-- Ò -->

```

```

<xs:enumeration value="211"/> <!-- Ó -->
<xs:enumeration value="212"/> <!-- Ô -->
<xs:enumeration value="213"/> <!-- Õ -->
<xs:enumeration value="214"/> <!-- Ö -->
<xs:enumeration value="215"/> <!-- × -->
<xs:enumeration value="216"/> <!-- Ø -->
<xs:enumeration value="217"/> <!-- Ù -->
<xs:enumeration value="218"/> <!-- Ú -->
<xs:enumeration value="219"/> <!-- Û -->
<xs:enumeration value="220"/> <!-- Ü -->
<xs:enumeration value="221"/> <!-- Ý -->
<xs:enumeration value="222"/> <!-- Þ -->
<xs:enumeration value="223"/> <!-- ß -->
<xs:enumeration value="224"/> <!-- à -->
<xs:enumeration value="225"/> <!-- á -->
<xs:enumeration value="226"/> <!-- â -->
<xs:enumeration value="227"/> <!-- ã -->
<xs:enumeration value="228"/> <!-- ä -->
<xs:enumeration value="229"/> <!-- å -->
<xs:enumeration value="230"/> <!-- æ -->
<xs:enumeration value="231"/> <!-- ç -->
<xs:enumeration value="232"/> <!-- è -->
<xs:enumeration value="233"/> <!-- é -->
<xs:enumeration value="234"/> <!-- ê -->
<xs:enumeration value="235"/> <!-- ë -->
<xs:enumeration value="236"/> <!-- ì -->
<xs:enumeration value="237"/> <!-- í -->
<xs:enumeration value="238"/> <!-- î -->
<xs:enumeration value="239"/> <!-- ï -->
<xs:enumeration value="240"/> <!-- ð -->
<xs:enumeration value="241"/> <!-- ñ -->
<xs:enumeration value="242"/> <!-- ò -->
<xs:enumeration value="243"/> <!-- ó -->
<xs:enumeration value="244"/> <!-- ô -->
<xs:enumeration value="245"/> <!-- õ -->
<xs:enumeration value="246"/> <!-- ö -->
<xs:enumeration value="247"/> <!-- ÷ -->
<xs:enumeration value="248"/> <!-- ø -->
<xs:enumeration value="249"/> <!-- ù -->
<xs:enumeration value="250"/> <!-- ú -->
<xs:enumeration value="251"/> <!-- û -->
<xs:enumeration value="252"/> <!-- ü -->
<xs:enumeration value="253"/> <!-- ý -->
<xs:enumeration value="254"/> <!-- þ -->
<xs:enumeration value="255"/> <!-- ÿ -->
</xs:restriction>
</xs:simpleType>

<xs:complexType name="ttsOptionsType">
  <xs:complexContent>

```

```

<xs:extension base="baseOptionsType">
  <xs:attribute name="adaptation" type="xs:string" use="optional"/>
  <xs:attribute name="autoclean_enabled" type="xs:boolean" use="optional"/>
  <xs:attribute name="autoclean_start_at" type="xs:string" use="optional"/>
  <xs:attribute name="enabled" type="xs:boolean" use="optional"/>
  <xs:attribute name="sign_cdr_enabled" type="xs:boolean" use="optional"/>
  <xs:attribute name="actual_calls_enabled" type="xs:boolean" use="optional"/>
  <xs:attribute name="actual_calls_archive_size" type="xs:integer" use="optional"/>
  <xs:attribute name="cdr_generation_enabled" type="xs:boolean" use="optional"/>
  <xs:attribute name="cdr_type" type="cdr_typeType" use="optional"/>
  <xs:attribute name="port_source" type="port_sourceType" use="optional"/>
  <xs:attribute name="cdr_time_rounding" type="cdr_time_roundingType"
use="optional"/>
  <xs:attribute name="cdr_field_delimiter" type="cdr_field_delimiterType"
use="optional"/>
  <xs:attribute name="manual_cdr_columns" type="xs:string" use="optional"/>
  <xs:attribute name="cdr_columns_name" type="xs:string" use="optional"/> <!--
Key:Value, Key2:Value2-->
  <xs:attribute name="partial_cdr_timeout" type="xs:integer" use="optional"/>
  <xs:attribute name="purge_older_than" type="xs:string" use="optional"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoInType"
minOccurs="1" maxOccurs="1" />
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- in -->

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainInfoOutType"
minOccurs="1" maxOccurs="1" />
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:choice>
  </xs:complexType>
</xs:element>

```

```
        </xs:all>
        </xs:complexType>
    </xs:element>
    <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
    </xs:choice>
</xs:complexType>
</xs:element>
<!-- out -->

</xs:schema>
```

Запрос:

http://192.168.1.21:9999/commands/tts_domain_options_info

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_domain_options_info.xsd">
  <request>
    <domain_options domain="biysk.local" />
  </request>
</in>
```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_domain_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="ttsOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="ttsOptionsType" sign_cdr_enabled="false" purge_older_than="2
month" port_source="iface_name" partial_cdr_timeout="300"
manual_cdr_columns=
"type,datetime,kod_a,kod_b,nai_a,nai_b,n_tr_gr_a,n_tr_gr_b,t_ecd,t_dba,cause_isup,cause_int,compl
etind,origin_dig,dialed_dig,actsusb,categ_a,place,rec_seq,server,serverind,treatment,conn_id,re
directed,redirection,orig_called,src_ip,dst_ip,port_a,port_b,diagnostic,signal_src_ip,signal_dst_
ip,signal_src_port,signal_dst_port,call_record_a,call_record_b" enabled="true"
cdr_type="extended" cdr_time_rounding="downside" cdr_generation_enabled="true"
cdr_field_delimiter="semicolon" cdr_columns_name="" autoclean_start_at="every 1 day at 00:00"
autoclean_enabled="true" adaptation="undefined" actual_calls_enabled="true"
actual_calls_archive_size="1000"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Нс_tts_domain_options_set

Установка значений свойств TTS на уровне домена.

Аналог команды в CoCon:

/domain/<DOMAIN>/tts/properties/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tts_domain_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа tts_domain_options_set.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="tts_domain_options_set">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Change TTS domain options.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <!--cdr_typeType-->
  <xs:simpleType name = "cdr_typeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="extended"/>
      <xs:enumeration value="normal"/>
      <xs:enumeration value="manual"/>
    </xs:restriction>
  </xs:simpleType>

  <!--port_sourceType-->
  <xs:simpleType name = "port_sourceType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="iface_name"/>
      <xs:enumeration value="iface_id"/>
    </xs:restriction>
  </xs:simpleType>

  <!--cdr_time_roundingType-->
  <xs:simpleType name = "cdr_time_roundingType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="downside"/>
      <xs:enumeration value="upside"/>
      <xs:enumeration value="downside_with_delta"/>
      <xs:enumeration value="upside_with_delta"/>
      <xs:enumeration value="mathematically"/>
      <xs:enumeration value="millisecond"/>
    </xs:restriction>
  </xs:simpleType>

  <!--cdr_field_delimiterType-->
  <xs:simpleType name = "cdr_field_delimiterType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="tab"/>
      <xs:enumeration value="space"/>
      <xs:enumeration value="comma"/>
      <xs:enumeration value="semicolon"/>
      <xs:enumeration value="8"/> <!-- ^H -->
    </xs:restriction>
  </xs:simpleType>

```



```
<xs:enumeration value="9"/> <!-- \t -->
<xs:enumeration value="10"/> <!-- \n -->
<xs:enumeration value="11"/> <!-- ^K -->
<xs:enumeration value="12"/> <!-- ^L -->
<xs:enumeration value="32"/> <!-- -->
<xs:enumeration value="33"/> <!-- ! -->
<xs:enumeration value="34"/> <!-- " -->
<xs:enumeration value="35"/> <!-- # -->
<xs:enumeration value="36"/> <!-- $ -->
<xs:enumeration value="37"/> <!-- % -->
<xs:enumeration value="38"/> <!-- & -->
<xs:enumeration value="39"/> <!-- ' -->
<xs:enumeration value="40"/> <!-- ( -->
<xs:enumeration value="41"/> <!-- ) -->
<xs:enumeration value="42"/> <!-- * -->
<xs:enumeration value="43"/> <!-- + -->
<xs:enumeration value="44"/> <!-- , -->
<xs:enumeration value="45"/> <!-- - -->
<xs:enumeration value="46"/> <!-- . -->
<xs:enumeration value="47"/> <!-- / -->
<xs:enumeration value="48"/> <!-- 0 -->
<xs:enumeration value="49"/> <!-- 1 -->
<xs:enumeration value="50"/> <!-- 2 -->
<xs:enumeration value="51"/> <!-- 3 -->
<xs:enumeration value="52"/> <!-- 4 -->
<xs:enumeration value="53"/> <!-- 5 -->
<xs:enumeration value="54"/> <!-- 6 -->
<xs:enumeration value="55"/> <!-- 7 -->
<xs:enumeration value="56"/> <!-- 8 -->
<xs:enumeration value="57"/> <!-- 9 -->
<xs:enumeration value="58"/> <!-- : -->
<xs:enumeration value="59"/> <!-- ; -->
<xs:enumeration value="60"/> <!-- < -->
<xs:enumeration value="61"/> <!-- = -->
<xs:enumeration value="62"/> <!-- > -->
<xs:enumeration value="63"/> <!-- ? -->
<xs:enumeration value="64"/> <!-- @ -->
<xs:enumeration value="65"/> <!-- A -->
<xs:enumeration value="66"/> <!-- B -->
<xs:enumeration value="67"/> <!-- C -->
<xs:enumeration value="68"/> <!-- D -->
<xs:enumeration value="69"/> <!-- E -->
<xs:enumeration value="70"/> <!-- F -->
<xs:enumeration value="71"/> <!-- G -->
<xs:enumeration value="72"/> <!-- H -->
<xs:enumeration value="73"/> <!-- I -->
<xs:enumeration value="74"/> <!-- J -->
<xs:enumeration value="75"/> <!-- K -->
<xs:enumeration value="76"/> <!-- L -->
<xs:enumeration value="77"/> <!-- M -->
```

```
<xs:enumeration value="78"/> <!-- N -->
<xs:enumeration value="79"/> <!-- O -->
<xs:enumeration value="80"/> <!-- P -->
<xs:enumeration value="81"/> <!-- Q -->
<xs:enumeration value="82"/> <!-- R -->
<xs:enumeration value="83"/> <!-- S -->
<xs:enumeration value="84"/> <!-- T -->
<xs:enumeration value="85"/> <!-- U -->
<xs:enumeration value="86"/> <!-- V -->
<xs:enumeration value="87"/> <!-- W -->
<xs:enumeration value="88"/> <!-- X -->
<xs:enumeration value="89"/> <!-- Y -->
<xs:enumeration value="90"/> <!-- Z -->
<xs:enumeration value="91"/> <!-- [ -->
<xs:enumeration value="92"/> <!-- \ -->
<xs:enumeration value="93"/> <!-- ] -->
<xs:enumeration value="94"/> <!-- ^ -->
<xs:enumeration value="95"/> <!-- _ -->
<xs:enumeration value="96"/> <!-- ` -->
<xs:enumeration value="97"/> <!-- a -->
<xs:enumeration value="98"/> <!-- b -->
<xs:enumeration value="99"/> <!-- c -->
<xs:enumeration value="100"/> <!-- d -->
<xs:enumeration value="101"/> <!-- e -->
<xs:enumeration value="102"/> <!-- f -->
<xs:enumeration value="103"/> <!-- g -->
<xs:enumeration value="104"/> <!-- h -->
<xs:enumeration value="105"/> <!-- i -->
<xs:enumeration value="106"/> <!-- j -->
<xs:enumeration value="107"/> <!-- k -->
<xs:enumeration value="108"/> <!-- l -->
<xs:enumeration value="109"/> <!-- m -->
<xs:enumeration value="110"/> <!-- n -->
<xs:enumeration value="111"/> <!-- o -->
<xs:enumeration value="112"/> <!-- p -->
<xs:enumeration value="113"/> <!-- q -->
<xs:enumeration value="114"/> <!-- r -->
<xs:enumeration value="115"/> <!-- s -->
<xs:enumeration value="116"/> <!-- t -->
<xs:enumeration value="117"/> <!-- u -->
<xs:enumeration value="118"/> <!-- v -->
<xs:enumeration value="119"/> <!-- w -->
<xs:enumeration value="120"/> <!-- x -->
<xs:enumeration value="121"/> <!-- y -->
<xs:enumeration value="122"/> <!-- z -->
<xs:enumeration value="123"/> <!-- { -->
<xs:enumeration value="124"/> <!-- | -->
<xs:enumeration value="125"/> <!-- } -->
<xs:enumeration value="126"/> <!-- ~ -->
<xs:enumeration value="160"/> <!-- -->
```

```
<xs:enumeration value="161"/> <!-- ¡ -->
<xs:enumeration value="162"/> <!-- ¢ -->
<xs:enumeration value="163"/> <!-- £ -->
<xs:enumeration value="164"/> <!-- ¤ -->
<xs:enumeration value="165"/> <!-- ¥ -->
<xs:enumeration value="166"/> <!-- ¦ -->
<xs:enumeration value="167"/> <!-- § -->
<xs:enumeration value="168"/> <!-- ¨ -->
<xs:enumeration value="169"/> <!-- © -->
<xs:enumeration value="170"/> <!-- ª -->
<xs:enumeration value="171"/> <!-- « -->
<xs:enumeration value="172"/> <!-- ¬ -->
<xs:enumeration value="173"/> <!-- ¯ -->
<xs:enumeration value="174"/> <!-- ® -->
<xs:enumeration value="175"/> <!-- ¯ -->
<xs:enumeration value="176"/> <!-- ° -->
<xs:enumeration value="177"/> <!-- ± -->
<xs:enumeration value="178"/> <!-- ² -->
<xs:enumeration value="179"/> <!-- ³ -->
<xs:enumeration value="180"/> <!-- ´ -->
<xs:enumeration value="181"/> <!-- µ -->
<xs:enumeration value="182"/> <!-- ¶ -->
<xs:enumeration value="183"/> <!-- · -->
<xs:enumeration value="184"/> <!-- , -->
<xs:enumeration value="185"/> <!-- ¹ -->
<xs:enumeration value="186"/> <!-- º -->
<xs:enumeration value="187"/> <!-- » -->
<xs:enumeration value="188"/> <!-- ¼ -->
<xs:enumeration value="189"/> <!-- ½ -->
<xs:enumeration value="190"/> <!-- ¾ -->
<xs:enumeration value="191"/> <!-- ¿ -->
<xs:enumeration value="192"/> <!-- À -->
<xs:enumeration value="193"/> <!-- Á -->
<xs:enumeration value="194"/> <!-- Â -->
<xs:enumeration value="195"/> <!-- Ã -->
<xs:enumeration value="196"/> <!-- Ä -->
<xs:enumeration value="197"/> <!-- Å -->
<xs:enumeration value="198"/> <!-- Æ -->
<xs:enumeration value="199"/> <!-- Ç -->
<xs:enumeration value="200"/> <!-- È -->
<xs:enumeration value="201"/> <!-- É -->
<xs:enumeration value="202"/> <!-- Ê -->
<xs:enumeration value="203"/> <!-- Ë -->
<xs:enumeration value="204"/> <!-- Ì -->
<xs:enumeration value="205"/> <!-- Í -->
<xs:enumeration value="206"/> <!-- Î -->
<xs:enumeration value="207"/> <!-- Ï -->
<xs:enumeration value="208"/> <!-- Ð -->
<xs:enumeration value="209"/> <!-- Ñ -->
<xs:enumeration value="210"/> <!-- Ò -->
```

```
<xs:enumeration value="211"/> <!-- Ó -->
<xs:enumeration value="212"/> <!-- Ô -->
<xs:enumeration value="213"/> <!-- Õ -->
<xs:enumeration value="214"/> <!-- Ö -->
<xs:enumeration value="215"/> <!-- × -->
<xs:enumeration value="216"/> <!-- Ø -->
<xs:enumeration value="217"/> <!-- Ù -->
<xs:enumeration value="218"/> <!-- Ú -->
<xs:enumeration value="219"/> <!-- Û -->
<xs:enumeration value="220"/> <!-- Ü -->
<xs:enumeration value="221"/> <!-- Ý -->
<xs:enumeration value="222"/> <!-- Þ -->
<xs:enumeration value="223"/> <!-- ß -->
<xs:enumeration value="224"/> <!-- à -->
<xs:enumeration value="225"/> <!-- á -->
<xs:enumeration value="226"/> <!-- â -->
<xs:enumeration value="227"/> <!-- ã -->
<xs:enumeration value="228"/> <!-- ä -->
<xs:enumeration value="229"/> <!-- å -->
<xs:enumeration value="230"/> <!-- æ -->
<xs:enumeration value="231"/> <!-- ç -->
<xs:enumeration value="232"/> <!-- è -->
<xs:enumeration value="233"/> <!-- é -->
<xs:enumeration value="234"/> <!-- ê -->
<xs:enumeration value="235"/> <!-- ë -->
<xs:enumeration value="236"/> <!-- ì -->
<xs:enumeration value="237"/> <!-- í -->
<xs:enumeration value="238"/> <!-- î -->
<xs:enumeration value="239"/> <!-- ï -->
<xs:enumeration value="240"/> <!-- ð -->
<xs:enumeration value="241"/> <!-- ñ -->
<xs:enumeration value="242"/> <!-- ò -->
<xs:enumeration value="243"/> <!-- ó -->
<xs:enumeration value="244"/> <!-- ô -->
<xs:enumeration value="245"/> <!-- õ -->
<xs:enumeration value="246"/> <!-- ö -->
<xs:enumeration value="247"/> <!-- ÷ -->
<xs:enumeration value="248"/> <!-- ø -->
<xs:enumeration value="249"/> <!-- ù -->
<xs:enumeration value="250"/> <!-- ú -->
<xs:enumeration value="251"/> <!-- û -->
<xs:enumeration value="252"/> <!-- ü -->
<xs:enumeration value="253"/> <!-- ý -->
<xs:enumeration value="254"/> <!-- þ -->
<xs:enumeration value="255"/> <!-- ÿ -->
</xs:restriction>
</xs:simpleType>

<xs:complexType name="ttsOptionsType">
  <xs:complexContent>
```

```

<xs:extension base="baseOptionsType">
  <xs:attribute name="autoclean_enabled" type="xs:boolean" use="optional"/>
  <xs:attribute name="autoclean_start_at" type="xs:string" use="optional"/>
  <xs:attribute name="adaptation" type="xs:string" use="optional"/>
  <xs:attribute name="enabled" type="xs:boolean" use="optional"/>
  <xs:attribute name="sign_cdr_enabled" type="xs:boolean" use="optional"/>
  <xs:attribute name="actual_calls_enabled" type="xs:boolean" use="optional"/>
  <xs:attribute name="actual_calls_archive_size" type="xs:integer" use="optional"/>
  <xs:attribute name="cdr_generation_enabled" type="xs:boolean" use="optional"/>
  <xs:attribute name="manual_cdr_columns" type="xs:string" use="optional"/>
  <xs:attribute name="cdr_columns_name" type="xs:string" use="optional"/><!--
Key:Value,Key2:Value2-->
  <xs:attribute name="cdr_type" type="cdr_typeType" use="optional"/>
  <xs:attribute name="port_source" type="port_sourceType" use="optional"/>
  <xs:attribute name="cdr_time_rounding" type="cdr_time_roundingType"
use="optional"/>
  <xs:attribute name="cdr_field_delimiter" type="cdr_field_delimiterType"
use="optional"/>
  <xs:attribute name="partial_cdr_timeout" type="xs:integer" use="optional"/>
  <xs:attribute name="purge_older_than" type="xs:string" use="optional"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetInType"
minOccurs="1" maxOccurs="1" />
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- in -->

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result">
        <xs:complexType>
          <xs:all>
            <xs:element name="domain_options" type="domainSetOutType"
minOccurs="1" maxOccurs="1" />

```

```

        </xs:all>
        </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
        </xs:choice>
    </xs:complexType>
</xs:element>
<!-- out -->

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/tts_domain_options_set

```

<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_domain_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="ttsOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="ttsOptionsType" purge_older_than="4 month"
partial_cdr_timeout="600" sign_cdr_enabled="false" />
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_domain_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="partial_cdr_timeout"/>
        <success option="purge_older_than"/>
        <success option="sign_cdr_enabled"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Ис_tts_domain_options_clean

Сброс значений свойств TTS уровня домена в значения по умолчанию.

Аналог команды в CoCon:

/domain/<DOMAIN>/tts/properties/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tts_domain_options_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа tts_domain_options_clean.xsd :

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="tts_domain_options_clean">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Reset TTS domain options to defaults.
      Usage: http://server:9999/commands/tts\_domain\_options\_clean
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="domain_options" type="domainCleanInType"
minOccurs="1" maxOccurs="1" />
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="domain_options" type="domainCleanOutType"
minOccurs="1" maxOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- out -->
</xs:schema>

```


Пример:

Запрос:

http://192.168.1.21:9999/commands/tts_domain_options_clean

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_domain_options_clean.xsd">
  <request>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <option name="purge_older_than"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ: 200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_domain_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="purge_older_than"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Нс_tts_ftp_domain_options - Управления настройками FTP-пользователя

- [Нс_tts_ftp_domain_options_info](#)
- [Нс_tts_ftp_domain_options_set](#)
- [Нс_tts_ftp_domain_optios_clean](#)

Нс_tts_ftp_domain_options_info

Просмотр пароля FTP-пользователя.

Аналог команды в CoCon:

/domain/<DOMAIN>/tts/ftp/properties/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tts_ftp_domain_options_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="tts_domain_options_info">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show TTS options.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <xs:complexType name="ttsFtpOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="password" type="nonEmptyTokenType" use="optional">
          <xs:annotation>
            <xs:documentation xml:lang="en">
              In case of strong security restrictions password may be shown as
              *****
            </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainInfoInType"
                minOccurs="1" maxOccurs="1" />
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">

```

```

        <xs:complexType>
            <xs:all>
                <xs:element name="domain_options" type="domainInfoOutType"
                    minOccurs="1" maxOccurs="1" />
            </xs:all>
        </xs:complexType>
    </xs:element>
    <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
</xs:choice>
</xs:complexType>
</xs:element>
<!-- out -->

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/tts_ftp_domain_options_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_ftp_domain_options_info.xsd">
  <request>
    <domain_options domain="biysk.local" />
  </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_ftp_domain_options_info.xsd">
  <result>
    <domain_options>
      <system>
        <options xs:type="ttsFtpOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="ttsFtpOptionsType" password="*****"/>
      </domain>
    </domain_options>
  </result>
</out>

```

Ис_tts_ftp_domain_options_set

Смена пароля FTP-пользователя на уровне домена.

Аналог команды в CoCon:

/domain/<DOMAIN>/tts/ftp/properties/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tts_ftp_domain_options_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="tts_ftp_domain_options_set">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Change FTP-user password.
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <xs:complexType name="ttsFtpOptionsType">
    <xs:complexContent>
      <xs:extension base="baseOptionsType">
        <xs:attribute name="password"
          type="nonEmptyTokenType" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainSetInType"
                minOccurs="1" maxOccurs="1" />
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result">
          <xs:complexType>
            <xs:all>
              <xs:element name="domain_options" type="domainSetOutType"
                minOccurs="1" maxOccurs="1" />
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- out -->

```

```

        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
    </xs:choice>
</xs:complexType>
</xs:element>
<!-- out -->

</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/tts_ftp_domain_options_set

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tts_ftp_domain_options_set.xsd">
  <request>
    <domain_options>
      <system>
        <options xs:type="ttsFtpOptionsType"/>
      </system>
      <domain name="biysk.local">
        <options xs:type="ttsFtpOptionsType" password="cdr4cdr"/>
      </domain>
    </domain_options>
  </request>
</in>

```

Ответ:200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_ftp_domain_options_set.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="password"/>
      </domain>
    </domain_options>
  </result>
</out>

```

hc_tts_ftp_domain_optios_clean

Сброс пароля доменного FTP-пользователя в значение по умолчанию.

Аналог команды в CoCon:

`/domain/<DOMAIN>/tts/ftp/properties/clean`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/tts_ftp_domain_optios_clean`

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="tts_domain_options_clean">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Reset TTS domain options to defaults.
      Usage: http://server:9999/commands/tts\_domain\_options\_clean
      Copyright (c) 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd" />
  <xs:include schemaLocation="ecm_common.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all minOccurs="1" maxOccurs="1">
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="domain_options" type="domainCleanInType"
minOccurs="1" maxOccurs="1" />
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="result" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:all minOccurs="1" maxOccurs="1">
              <xs:element name="domain_options" type="domainCleanOutType"
minOccurs="1" maxOccurs="1"/>
            </xs:all>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- out -->

</xs:schema>

```

Пример:

Запрос:

http://192.168.1.21:9999/commands/tts_ftp_domain_options_clean

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tts_ftp_domain_options_clean.xsd">
  <request>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <option name="password"/>
      </domain>
    </domain_options>
  </request>
</in>
```

Ответ:200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_ftp_domain_options_clean.xsd">
  <result>
    <domain_options>
      <system/>
      <domain name="biysk.local">
        <success option="password"/>
      </domain>
    </domain_options>
  </result>
</out>
```

Нс_tts - Управление CDR

- [Hc_tts_prefix](#)
- [Hc_tts_mode](#)
- [Hc_tts_add_cdr_group](#)
- [Hc_tts_change_cdr_group](#)
- [Hc_tts_show_cdr_group](#)
- [Hc_tts_info](#)
- [Hc_tts_remove_cdr_group](#)
- [Hc_rename_cdr_group](#)
- [Hc_tts_validate_cdr](#)

Нс_tts_prefix

Команда позволяет установить префикс по умолчанию для всех CDR-файлов, которые будут генерироваться системой.

Аналог команды в CoCon:

`/domain/<domain>/cdr/prefix`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/tts_prefix`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить двумя сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="prefix" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="domain" type="domainType" minOccurs="1" maxOccurs="1" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: `http://192.168.1.21:9999/commands/tts_prefix`

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tts_prefix.xsd">
  <domain name="biysk.local" prefix="cdr_bsk" />
</in>
```

Ответ: 201

[Hc_tts_mode](#)

Команда позволяет установить режим генерации файла CDR-группы .default.

Аналог команды в CoCon:

/domain/<domain>/cdr/mode

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tts_mode

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- valueType -->
  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:integer" use="required"/>
  </xs:complexType>
  <!-- time_of_dayType -->
  <xs:complexType name="time_of_dayType">
    <xs:attribute name="hours" type="xs:integer" use="required"/>
    <xs:attribute name="minutes" type="xs:integer" use="required"/>
    <xs:attribute name="seconds" type="xs:integer" use="required"/>
  </xs:complexType>
  <!-- modeType -->
  <xs:complexType name="modeType">
    <xs:choice>
      <xs:element name="count" type="valueType"/>
      <xs:element name="period" type="valueType"/>
      <xs:element name="time_of_day" type="time_of_dayType" maxOccurs="unbounded"/>
    </xs:choice>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="mode" type="modeType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/tts_mode

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="tts_mode.xsd">
  <mode domain="biysk.local">
    <time_of_day hours="12" minutes="00" seconds="00"/>
    <time_of_day hours="23" minutes="59" seconds="59"/>
  </mode>
</in>

```

Ответ: 201

Нс_tts_add_cdr_group

Команда позволяет создать новую CDR-группу.

Аналог команды в CoCon:

`/domain/<domain>/cdr/add_cdr_group`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/tts_add_cdr_group`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить двумя сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:integer" use="required"/>
  </xs:complexType>
  <xs:complexType name="time_of_dayType">
    <xs:attribute name="hours" type="xs:integer" use="required"/>
    <xs:attribute name="minutes" type="xs:integer" use="required"/>
    <xs:attribute name="seconds" type="xs:integer" use="required"/>
  </xs:complexType>
  <xs:simpleType name="typeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="time"/>
      <xs:enumeration value="period"/>
      <xs:enumeration value="count"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="formatType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="csv"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="formatsType">
    <xs:sequence>
      <xs:element name="format" maxOccurs="4">
        <xs:complexType>
          <xs:attribute name="value" type="formatType" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="cdrGroupType">
    <xs:sequence>
      <xs:choice>
        <xs:element name="count" type="valueType"/>
        <xs:element name="period" type="valueType"/>
        <xs:element name="time_of_day" type="time_of_dayType" maxOccurs="unbounded"/>
      </xs:choice>
      <xs:element name="formats" type="formatsType"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="typeType" use="required"/>
    <xs:attribute name="prefix" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="domainType">
    <xs:all>
      <xs:element name="cdr_group" type="cdrGroupType"/>
    </xs:all>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
</xs:schema>

```

```

</xs:complexType>
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="domain" type="domainType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/tts_add_cdr_group

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tts_add_cdr_group.xsd">
  <domain name="biysk.local">
    <cdr_group name="group6" type="time" prefix="">
      <time_of_day hours="12" minutes="30" seconds="45" />
      <time_of_day hours="20" minutes="10" seconds="05" />
      <formats>
        <format value="csv" />
      </formats>
    </cdr_group>
  </domain>
</in>

```

Ответ: 201

[Hc_tts_change_cdr_group](#)

Команда позволяет изменять параметры CDR-группы.

Аналог команды в CoCon:

/domain/<domain>/cdr/change_cdr_group

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tts_change_cdr_group

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 201 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="error.xsd"/>

  <!-- types -->
  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:integer" use="required"/>
  </xs:complexType>

  <xs:complexType name="time_of_dayType">
    <xs:attribute name="hours" type="xs:integer" use="required"/>
    <xs:attribute name="minutes" type="xs:integer" use="required"/>
    <xs:attribute name="seconds" type="xs:integer" use="required"/>
  </xs:complexType>

  <xs:simpleType name="typeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="time"/>
      <xs:enumeration value="period"/>
      <xs:enumeration value="count"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="formatType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="csv"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="formatsType">
    <xs:sequence>
      <xs:element name="format" maxOccurs="4">
        <xs:complexType>
          <xs:attribute name="value" type="formatType" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="cdrGroupType">
    <xs:sequence>
      <xs:choice>
        <xs:element name="count" type="valueType"/>
        <xs:element name="period" type="valueType"/>
        <xs:element name="time_of_day" type="time_of_dayType" maxOccurs="unbounded"/>
      </xs:choice>
      <xs:element name="formats" type="formatsType"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

```

```

    <xs:attribute name="type" type="typeType" use="required"/>
    <xs:attribute name="prefix" type="xs:string" use="required"/>
    <xs:attribute name="new_name" type="xs:string"/>
</xs:complexType>

<xs:complexType name="domainType">
  <xs:choice>
    <xs:element name="cdr_group" type="cdrGroupType"/>
  </xs:choice>
  <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>

<xs:simpleType name="resultType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="ok"/>
    <xs:enumeration value="domain_not_exists"/>
    <xs:enumeration value="group_not_exists"/>
  </xs:restriction>
</xs:simpleType>
<!-- types end -->

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="domain" type="domainType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- /in -->

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="response" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:attribute name="result" type="resultType" use="required"/>
        </xs:complexType>
      </xs:element>
      <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
<!-- /out -->

</xs:schema>

```

Запрос: http://192.168.1.21:9999/commands/tts_change_cdr_group

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_change_cdr_group.xsd">
  <domain name="biysk.local">
    <cdr_group name="group6" type="period" prefix="g222" >
      <period value="250" />
      <formats>
        <format value="csv" />
      </formats>
    </cdr_group>
  </domain>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_change_cdr_group.xsd">
  <response result="ok"/>
</out>
```

[Hc_tts_show_cdr_group](#)

Команда позволяет просмотреть CDR-группы, декларированные в системе.

Аналог команды в CoCon:

`/domain/<domain>/cdr/show`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/tts_show_cdr_group`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="cdr_group" type="xs:string" use="optional"/>
  </xs:complexType>
  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:integer" use="required"/>
  </xs:complexType>
  <xs:complexType name="time_of_dayType">
    <xs:attribute name="hours" type="xs:integer" use="required"/>
    <xs:attribute name="minutes" type="xs:integer" use="required"/>
    <xs:attribute name="seconds" type="xs:integer" use="required"/>
  </xs:complexType>
  <xs:simpleType name="formatType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="csv"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="formatsType">
    <xs:sequence>
      <xs:element name="format" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="value" type="formatType" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="typeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="time"/>
      <xs:enumeration value="period"/>
      <xs:enumeration value="count"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="cdrGroupType">
    <xs:sequence>
      <xs:choice>
        <xs:element name="count" type="valueType"/>
        <xs:element name="period" type="valueType"/>
        <xs:element name="time_of_day" type="time_of_dayType" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:choice>
      <xs:element name="formats" type="formatsType" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="typeType" use="required"/>
    <xs:attribute name="prefix" type="xs:string" use="required"/>
  </xs:complexType>

```

```

<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="domain" type="domainType"/>
    </xs:all>
  </xs:complexType>
</xs:element>
<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="cdr_group" type="cdrGroupType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/tts_show_cdr_group

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tts_show_cdr_group.xsd">
  <domain name="biysk.local" cdr_group="group6"/>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tts_show_cdr_group.xsd">
  <cdr_group name="group6" type="period" prefix="g222">
    <period value="250"/>
    <formats>
      <format value="csv"/>
    </formats>
  </cdr_group>
</out>

```

Запрос: http://192.168.1.21:9999/commands/tts_show_cdr_group

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tts_show_cdr_group.xsd">
  <domain name="biysk.local"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tts_show_cdr_group.xsd">
  <cdr_group name="bsk3" type="time" prefix="bsk3">
    <time_of_day hours="0" minutes="0" seconds="0"/>
    <formats>
      <format value="csv"/>
    </formats>
  </cdr_group>
  <cdr_group name="group6" type="period" prefix="g222">
    <period value="250"/>
    <formats>
      <format value="csv"/>
    </formats>
  </cdr_group>
</out>
```

[Hc_tts_info](#)

Команда позволяет просмотреть настройки группы .default.

Аналог команды в CoCon:

/domain/<domain>/cdr/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tts_info

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- valueType -->
  <xs:complexType name="valueType">
    <xs:attribute name="value" type="xs:integer" use="required"/>
  </xs:complexType>
  <!-- prefixType -->
  <xs:complexType name="prefixType">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- time_of_dayType-->
  <xs:complexType name="time_of_dayType">
    <xs:attribute name="hours" type="xs:integer" use="required"/>
    <xs:attribute name="minutes" type="xs:integer" use="required"/>
    <xs:attribute name="seconds" type="xs:integer" use="required"/>
  </xs:complexType>
  <!-- formatType-->
  <xs:simpleType name="formatType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="csv"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- formatsType-->
  <xs:complexType name="formatsType">
    <xs:sequence>
      <xs:element name="format" maxOccurs="4">
        <xs:complexType>
          <xs:attribute name="value" type="formatType" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <!-- domainType-->
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- infoType-->
  <xs:complexType name="infoType">
    <xs:all>
      <xs:element name="prefix" type="prefixType" minOccurs="0"/>
      <xs:element name="formats" type="formatsType" minOccurs="0"/>
      <xs:element name="mode" minOccurs="0">
        <xs:complexType>
          <xs:choice>
            <xs:element name="count" type="valueType" minOccurs="0"/>
            <xs:element name="period" type="valueType" minOccurs="0"/>
            <xs:element name="time_of_day" type="time_of_dayType" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:choice>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>

```



```

        </xs:complexType>
    </xs:element>
</xs:all>
</xs:complexType>
<!-- in -->
<xs:element name="in">
    <xs:complexType>
        <xs:all>
            <xs:element name="domain" type="domainType"/>
        </xs:all>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="info" type="infoType" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример:

Запрос: http://192.168.1.21:9999/commands/tts_info

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tts_info.xsd">
    <domain name="biysk.local" />
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out
    xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tts_info.xsd">
    <info>
        <prefix value="cdr_bsk"/>
        <mode>
            <time_of_day hours="12" minutes="0" seconds="0"/>
            <time_of_day hours="23" minutes="59" seconds="59"/>
        </mode>
    </info>
</out>

```

Нс_tts_remove_cdr_group

Команда позволяет удалить CDR-группу из системы.

Аналог команды в CoCon:

`/domain/<domain>/cdr/remove`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/tts_remove_cdr_group`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить следующими сообщениями:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="domainType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="cdr_group" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="domain" type="domainType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос: `http://192.168.1.21:9999/commands/tts_remove_cdr_group`

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="tts_remove_cdr_group.xsd">
  <domain name="biysk.local" cdr_group="gs_test3" />
</in>
```

Ответ: 204

[Hc_rename_cdr_group](#)

Команда позволяет переименовать группу CDR

Аналог команды в CoCon:

`domain/p.city/cdr/rename_cdr_group`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

`http://host:port/commands/tts_rename_cdr_group`

Код HTTP-ответа:

При выполнении команды HTTP-терминал может ответить двумя сообщениями:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла с ответом `rename_cdr_group.xsd`:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="rename_cdr_group">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Rename CDR group
      Copyright (C) 2015, Eltex. All right reserved.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="error.xsd"/>

  <xs:simpleType name="resultType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="ok"/>
      <xs:enumeration value="domain_not_exists"/>
      <xs:enumeration value="group_not_exists"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="old_name" type="xs:string" use="required"/>
            <xs:attribute name="new_name" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- /in -->

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="result" type="resultType" use="required"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType" minOccurs="1" maxOccurs="1"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <!-- /out -->

```

```
</xs:schema>
```

Пример

Запрос:

http://192.168.1.21:9999/commands/tts_rename_cdr_group

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="rename_cdr_group.xsd">
  <request domain="biysk.local" old_name="group6" new_name="gs_test3"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="rename_cdr_group.xsd">
  <response result="ok"/>
</out>
```

[Hc_tts_validate_cdr](#)

Команда позволяет проверить подпись CDR файла или определенной записи в CDR.

Аналог команды в CoCon:

`domain/<DOMAIN>/cdr/validate`

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/tts_validate_cdr

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Common error message for HttpTerminal application.
      Copyright 2015, Eltex. All rights reserved.

      Command to check signature on the CDR file or specific line on the CDR file.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <!-- cdrTypeType -->
  <xs:simpleType name="cdrTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="line"/>
      <xs:enumeration value="file"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- cdrType -->
  <xs:complexType name="cdrType">
    <xs:attribute name="type" type="cdrTypeType" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          line - request on validation specific line of the CDR file;
          file - request on validation whole CDR file
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="content" type="xs:base64Binary" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          Base64 encoded content (CDR file or one line of CDR file) to check signature.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
  <!-- requestType -->
  <xs:complexType name="requestType">
    <xs:all>
      <xs:element name="cdr" type="cdrType"/>
    </xs:all>
    <xs:attribute name="domain" type="xs:string" use="required"/>
  </xs:complexType>
  <!-- responseType -->
  <xs:complexType name="responseType">
    <xs:attribute name="valid" type="xs:boolean" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="en">

```

```

        true - in case of entered CDR file of CDR line is valid;
        false - otherwise
    </xs:documentation>
</xs:annotation>
</xs:attribute>
</xs:complexType>
<!-- in -->
<xs:element name="in">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="request" type="requestType"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="response" type="responseType"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Пример

Запрос: http://192.168.1.21:9999/commands/tts_validate_cdr

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="tts_validate_cdr.xsd">
    <request domain="biysk.local">
        <cdr type="line" content="Y2RyIGxpbmU="/>
    </request>
</in>

```

Ответ:

```

<?xml version="1.0"?>
<out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="tts_validate_cdr.xsd">
    <response valid="false"/>
</out>

```

Запрос на валидацию всего файла (параметр cdr type="file") http://192.168.1.21:9999/commands/tts_validate_cdr:

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="tts_validate_cdr.xsd">
  <request domain="biysk.local">
    <cdr type="file"
content="VF1QRTtEQVRJTUU7S09EX0E7S09EX0I7T19UU19HU19B005fVFJfR1JfQjtUX0VDRD
tUX0RCQTtU
X0ROQTtDQVVTRV9JU1VQ00NBVVNFx010VDtDT01QTEVUSU5E00RJQUxFRF9ESUc7QUNUU1VTQjt
D
QVRFR19B01BMQUNF01JFQ19TRVE7U0VSV1VTRVI7U0VSVk10RE1D01RSRUFUTUV0VDtDT050X01
E
01JFRE1SRUNURUQ7UkVESVJFQ1RJT047T1JJR19DQUxMRUQ7U1JDX01Q00RTVF9JUDtQT1JUX0E
7
UE9SVF9C001FRE1BX1BST1hZ00RJQUd0T1NUSUM7Q09ERUM7U01HTgpDYWxs0zIwMTkuMDYuMDY
g
MDc6MzU6MDA7MjQwNDcw0zI0MDEwMTs70zI70Tsw0zE2025vcm1hbDs30zI0MDEwMTs7MTA7Mjs
w
0zI0MDQ3MDS7Q2FsbCBpcyBjb21wbGV0ZTszNzYxMTI5NTMwMjA5Njc3NDEx0zs70zsx0TIuMTY
4
LjIuMjY7c3lzdGVtOm12cjswNjE2ZTFjZTRiNzg2YWYx0zE70zs2ZldMRnMrNEFuR2tvRjNuTE8
w
ay9iYXhGbzh2N1FUeW0zUmgreDdIMWZxWmpuaFArTUxSVk9xVUhXVldIRWUzVXgvUW9yYVFKVnk
r
dWOWTEpJ0ThiUT09C1N1cnZpY2U7MjAxOS4wNi4wNiAwNzozNTowMDSyNDA0NzA7MjQwMTAx0zs
7
MDsw0zA7MDS70ztpbnZvY2F0aW9u0zs7MTsyNDAxMDE7Y2FsbF9yZWVvcnRpbmc70zM3NjExMjk
1
MzAyMDk2Nzc0MTE70zs70zs70zs700Y2dm52NEgxV1NMenZnM2dRc0FxrNptWmpoaC83SGNmRWg
1
K3VFS3lkNnNvaVg5MkwrS1Q3T2lxdUNBaDQxMTNSTi9TQ0dXaG5HcjI5T3hkTVdSYkhNPT0KQ2F
S
bDsyMDE5LjA2LjA2IDA30jM10jEx0zI0MDQ3MDSyNDA0NjQ70zsy0zE7MDSxNjtub3JtYWw7Nzs
y
NDA0NjQ70zEw0zI7MDSyNDA0NzA700NhbGwgaXMgY29tcGxldGU7Nz3Nzc50DAzMTM4Mzg20DQ
2
OTs70zs7MTkyLjE2OC4yLjIwMDtzeXN0ZW06aXZy0zA2MTZlMWNkZjg5YmE20DQ7MTs70y9Rb3p
a
WHVBNThMznp1Vno2awZ0TH1iL2xpT3hYNHZHcwoxQmdB0W1E0GYvR3BpZWlLQnU4L1pwNVpTamd
T
ZGc4e1duV291czFLRDZDK3BMTEUyVnd3PT0KU2Vydm1jZTsyMDE5LjA2LjA2IDA30jM10jEx0zI
0
MDQ3MDSyNDA0NjQ70zsw0zA7MDsw0zs702ludm9jYXRpb2470zsw0zI0MDQ2NDtjYWxsX3JlY29
y
ZGluZzs7Nz3Nzc50DAzMTM4Mzg20DQ20Ts70zs70zs70zs70Xh0TTN3RDnuY25yUDZGdEFYRnp
j
bWR1NFdHbkJMR3JkQjF6Rms2SS9raVVoMWQ3azM2NG1HRG1LeXpkTVJ0Ykp4UTFzaGZPRVZaT2R
G
MS90WDBQRUE9PQpDYWxs0zIwMTkuMDYuMDYgMDC6MzU6MTU7Mzg1NDI0MDQ3MDS0MTY5Nzc702J
```


p
eXNrLmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTsyOzU7MDsxNjtub3JtYww7Nzs0MTY5Nzc
7
OzEwOzA7MDszODU0MjQwNDcwOztCeSBpdnIuIFNjcm1wdCBoYXMGYmVlbiBjb21wbGV0ZWQgYnk
g
ZGVmYXVsdC47MzcwMjxMzUwNjkyMzc4NzA1Nzs70zs7MTkyLjE2OC4yLjg7c3lzdGVtOm12cjt
z
bwctNDsx0zs7VkhzOXk3M11MU1NsZBSd0d4K0RGVklkdklrS2dwTW5NemNEdFJNQ3p5ZmJoYk1
o
KzduR1JieEJSU0I4bHR5aDQ5RXZWMc9GejNHV1NlQzJES25S0Xc9PQpDYWxsOzIwMTkuMDYuMDY
g
MDg6MDA6MDA7Mzg1NDI0MDQ3MDswNTg702JpeXNrLmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmF
t
ZTsyOzU7MDsxNjtub3JtYww7NzswNTg7OzEwOzA7MDszODU0MjQwNDcwOztCeSBpdnIuIFNjcm1
w
dCBoYXMGYmVlbiBjb21wbGV0ZWQgYnkGZGVmYXVsdC47MzQ3MzE3MzgyOTI10TkxMTQ3Mzs70zs
7
MTkyLjE2OC4yLjg7c3lzdGVtOm12cjtzbwctNDsx0zs7eFVaTEQzctVPU1k1bkU1Rng2MTkyVWJ
z
ZUora0NHRkpDZW1rTmRYTTNRUhh6djJYeEiybzNzZn1LYXRYRWQ3REN4RDRUZVdrUXk2SmhMck5
Q
YVJKMVE9PQpDYWxsOzIwMTkuMDYuMDYgMDg6MDA6MDg7MjQwNDcwOzI0MDEwMTs70zM7NzswOZE
2
025vcm1hbDs30zI0MDEwMTs7MTA7MjswOzI0MDQ3MDs7Q2FsbCBpcyBjb21wbGV0ZTs0MDQ5NjM
2
Nzk0MzYzNDkxODk2Ozs70zsw0TIuMTY4LjIuMjY7c3lzdGVtOm12cjswnjE2ZTFjZTRiNzg2YWY
x
OzE7OzthY1drCVRqZ3hWT2FTSmNnNlmeVU20DlJbjRTZUSWEM0nkVQVS900E5xS2N4d1I4SUE
r
YwVvc21wLy9Sb3gyQ0NEUGR30GZSS3F0ajN6Zm12WVBoZz09ClNlcnZpY2U7MjAx0S4wNi4wNiA
w
ODowMDowODsyNDA0NzA7MjQwMTAx0zs7MDswOzA7MDs70ztpbnZvY2F0aW9uOzs7MTsyNDAxMDE
7
Y2FsbF9yZWVvcnRpbmc7OzQwNDk2MzY3OTQzNjM00TE40TY70zs70zs70zs70096bGxyRG5TTU9
n
RS9SUTk5RGRURHpwUHV2Qm12cFJaSm5QbkZFZG0rN2Yvd3ZsRVd4TjcyRklnWEFlc2gxOUhkeGw
0
wnNoZ1MwRU03RFYrdkxEdWZRPT0KQ2FsbDsyMDE5LjA2LjA2IDA40jAw0jE40zI0MDQ3MDsyNDA
0
NjQ7OzsyOzE7MDsxNjtub3JtYww7NzsyNDA0NjQ7OzEwOzI7MDsyNDA0NzA700NhbGwgaXMgY29
t
cGxldGU7MzYx0TAzMDYzNTE2MzkwMjc2ODs70zs7MTkyLjE2OC4yLjIwMDtzeXN0ZW06aXZyOzA
2
MTZlMWNkZjg5YmE2ODQ7MTs700JnYk04QisyYjd5b0tGV3FKQ3ZDMjhMdUt1Nk9JUWxIQmFKbHN
Z
U3BVWThPRW0vMEY3Wk9sTVM4QklkWTVfd1R0S1crK0RGUFVBRGwvndNuaDJ0ZGd3PT0KU2Vydm1
j
ZTsyMDE5LjA2LjA2IDA40jAw0jE40zI0MDQ3MDsyNDA0NjQ7OzswOzA7MDswOzs7021udm9jYXR
p

b2470zsx0zI0MDQ2NDtjYwxsX3JlY29yZGluZzs7MzYx0TAzMDYzNTE2MzkwMjc20Ds70zs70zs
 7
 0zs7T2N0YXBSRlMwNEVNanpXRzFNUkVQMxoyUHNknjRhWkF0MVh1SndNU1RkUkpkQ0hpd0F0T0F
 K
 Tng2Y1JYQXhzQnN0ZTZKeEphc2JEUETlY1pPbVcxWgc9PQpDYWxs0zIwMTkuMDYuMDYgMDg6MzI
 6
 MDY70TYxNjI2NjE00TsyNDUyODU7Ym15c2subG9jYwuc21nLmdyLnRydW5rLmF1dG9uYW1102J
 p
 eXNrLmxvY2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTsw0zA7MjQ7MTAy025vcm1hbDsx0zI0NTI
 4
 NTs7MTA7MTsw0zk2MTYyNjYxNDk700NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjsxNDM
 1
 MTI0MjQ20zs70zE5Mi4xNjguMi440ztzbwctNDt1bXMx0zA701BDTUESUENNVsXHNzI5LHR1bGV
 w
 aG9uZS1ldmVudDtPSDhUN3RBL21SR0hwZ2RLc2tMR1h1eUJv0ThQZkF1ZWtzcFVUeDdBYUdYZXR
 u
 MlppVGRJdWljY3AzUXVLYmN4NWNpVl0VnVqUeTGymVkSDUyUWE0QT09CkNhbGw7MjAx0S4wNi4
 w
 NiAw0Do0MT0xMjs5NjE50TE40Tg50zI0MDUwNztiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV
 0
 b25hbWU70zMwMDsx0zA7MDs7NzsyNDA1MDC70zEw0zsx0zk2MTk5MTg50Dk70zszNzI1NTAyNzQ
 5
 0zs70zE5Mi4xNjguMi440zE5Mi4xNjguMi4yNjtzwbwctNDswNjE2ZTFjZTA0Mzgz0Dk20zE701B
 D
 TUESdGVsZXBob251LWV2ZW5003R2T1dyMlQ0b3hCNElxbTRIqj1HVHBxRVVNaHNYMHU4YVVMUjR
 B
 M0R6MDFIMlpYdDFTakIwNkduWjNtUGNTWjdHdEQ2RF16Swx1SzU0S21GZ083ZkR3PT0KQ2FsbDs
 y
 MDE5LjA2LjA2IDA40jQ20jE00zk2MTk5MTg50Dk7MjQwNTA302JpeXNrLmxvY2FsLnNtZy5nci5
 0
 cnVuay5hdXRvbmFtZTs7MzAw0zA7MDsw0zs30zI0MDUwNzs7MTA70zI70TYx0Tkx0Dk40Ts70zM
 3
 MjU1MDI3NDk70zs7MTkyLjE20C4yLjg7MTkyLjE20C4yLjI203NtZy000zA2MTZlMWNlMDQzODM
 4
 OTY7MTs7UENNSx0ZWxlGhvbUUtZXZlbnQ7SytyZnN2SHFDU3diRHh0UzNh0XdncE9sNkUwYjJ
 u
 azB60Vd1SXNLRV1INVNTEhVlZk1pZF16Ky9VY0c0L2V0TERlUytrRRDQxUGNXZTRUuzEyNGc1M3c
 9
 PQpDYWxs0zIwMTkuMDYuMDYgMDg6NTE6MTQ70TYx0Tkx0Dk40TsyNDA1MDC7Ym15c2subG9jYw
 u
 c21nLmdyLnRydW5rLmF1dG9uYW110zszMDA7MDsw0zA70zc7MjQwNTA30zszMDs7Mzs5NjE50TE
 4
 OTg50zs7MzcyNTUwMjc00Ts70zsx0TIuMTY4LjIu0Ds0TIuMTY4LjIuMjY7c21nLTQ7MDYxNmU
 x
 Y2UwNDM4Mzg5Njxs0ztQQ01BLHR1bGVwaG9uZS1ldmVudDtwc1krS1hTTG11TlduZjl2SDBMe1A
 z
 OWFwZDJSNcyw1o5SUTjQ3phT1pjdmdkazMvRGtJazhoaUN5bGgxm5NYk1TU1FONHD4azV5M21
 1

Ozsx0TIuMTY4LjIuODsx0TIuMTY4LjIuMjY7c21nLTQ7MDYxNmUxY2UwNDM4Mzg5Nj5x0ztQQ01
B
LHR1bGVwaG9uZS1ldmVudDtlldisydnE2cGR1TFErZHhwV1d5eHZRZFN1ak1qK09HU1NFUkw00Gd
O
w1Q4eUV4dWE1TtWcGFJTXZhczhINlJrcUVzYnI4KzdwU2JaRnBPU0hWS0RGQT09CkNhbGw7MjA
X
OS4wNi4wNiAw0ToxNjoxNDS5NjE5OTE4OTg5OzI0MDUwNztiaXlZay5sb2NhbC5zbWcuZ3IudHJ
1
bmsuYXV0b25hbWU7OzE3ODswOzA7MTY7bm9ybWFsOzc7MjQwNTA3Oz5xMDsxOzA7OTYxOTkxODk
4
OTs7SVNVUCBjYwxsIGlzIGNvbXBsZXRL0zM3MjU1MDI3NDk7Ozs7MTkyLjE2OC4yLjg7MTkyLjE
2
OC4yLjI2O3NtZy00OzA2MTZlMWNlMDQzODM4OTY7MTs7UENNQsX0ZwXlcGhvbmUtZXZlbnQ7UTV
Z
VzE1Mkc1TTFQd1V1bW1XbGRwMUNBdzY2cFcrWEowSEFsR1lJQjcwampFUGFhdmZVN2pHWUtUN2V
O
Z3pDdzFaMWJYQkVXVnNQWmJ5a0pUMkVyQmc9PQpDYwxsOzIwMTkuMDYuMDYgMDk6MjY6MTE7OTY
1
MDI1MjY5NDsyNDUyODU7Ym15c2subG9jYwwuc21nLmdyLnRydW5rLmF1dG9uYW1lO2JpeXNrLmx
v
Y2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTswOzA7MjQ7MTAy025vcm1hbDsxOzI0NTI4NTs7MTA
7
MTswOzk2NTAyNTI2OTQ7O0NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjsy0TE1MTgzNTQ
X
Ozs7OzE5Mi4xNjguMi44OztzbWctNDtlbXMX0zA7O1BDTUESUENNVSxHNzI5LHR1bGVwaG9uZS1
1
dmVudDtpTHJ2dVB2UW5TajE4Z0ErS3Z0THpVcUdHNW9oMVhwQXhVMmwwcmJlDtdBT1V0dDk4YjB
F
SytEMWR6ZzB0VS82bXRqZE12UWZBRk5vdENLMEpjVG9IQT09CkNhbGw7MjAxOS4wNi4wNiAw0To
1
MzowNDszODUyMjYxNzAwOzI0NTk2MjYxNzAwOzI0NTk2MjYxNzAwOzI0NTk2MjYxNzAwOzI0NTk2
7
Ym15c2subG9jYwwuc3N3LmdyLnRydW5rLmF1dG9uYW1lOzA7MDsyNTsxODtub3RSZWFjaGFibGU
7
MTsyNDU5NjI7OzEwOzA7MDszODUyMjYxNzAwOzI0NTk2MjYxNzAwOzI0NTk2MjYxNzAwOzI0NTk2
Z
ZTszMzYzOTY1MjU7Ozs7MTkyLjE2OC4yLjg7O3NtZy00O2VtczE7MDs7UENNQsXQ001VLEc3Mjk
S
dGVsZXBob25lLWV2ZW50O1ZuVWpjUHVQWXT0HRpamFOUDlKd2xCWwswZVJxQzFBSGJMbd1UL2R
W
wVZ6Um4yU0gzMmp2U2haTGNscFJmNkNqK2RCwVFIUES0NFBGTFftVTJYckdBPT0KQ2FsbDsyMDE
5
LjA2LjA2IDEwOjAwOjQ2OzkwOTQ1NDMwMjY7MjY1MjY1MjY1MjY1MjY1MjY1MjY1MjY1MjY1MjY1
u
ay5hdXRvbmFtZTtiaXlZay5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV0b25hbWU7MDswOzI0OzMX025
v
cm1hbDsxOzI0NTI4NTs7MTA7MTswOzkwOTQ1NDMwMjY7O0NhbGwgaXMgY2FuY2VsbGVkIGJ5IGl
u

aXRpYXRvcjsyMjczNzc4MjA2Ozs7OzE5Mi4xNjguMi440ztzbwctNDt1bXMxOzA701BDTUESUEN
N
VSxHNzI5LHR1bGVwaG9uZS1ldmVudDs1Q0FOMk9YUVdZRkswTX11NzBwcXk3VXpYeS9NcTdxWXh
K
WXgvYzNHbzUyRUFhSVVxeHpjMD1vcTdIUjUxwXpYR1EwcXRhSG9RM016UGRLNGdIU2JqZz09CkN
h
bGw7MjAxOS4wNi4wNiAxMDoxMjo0Njs5NjQ1MTM1MzM4OzI0NTk40DtiaXlZay5sb2NhbC5zbWc
u
Z3IudHJ1bmsuYXV0b25hbWU7Ym15c2subG9jYwwuc3N3LmdyLnRydW5rLmF1dG9uYW11OzA7MDs
y
NDsxMDI7bm9ybWFsOzE7MjQ10Tg40zsxMDsxOzA70TY0NTEzNTMzODs7Q2FsbCBpcyBjYW5jZWx
s
ZWQgYnkgaW5pdG1hdG9yOzk5ODgyNjM0NTs7OzsxOTIuMTY4LjIuODs7c21nLTQ7ZW1zMTswOzt
Q
Q01BLFBDTVUsRzcyOSx0ZWxlGhvbmtZlbnQ7czF0FmxdmZNTTlPM1hBdUlQTnUwS3VNZUt
V
wVziYmZuSU5NMEVCdS9xeXVGRGd4dHJ3dW11VUVaTzVNB2hzQVViUDFIeXFia3A1Z1NBenc3WEV
Y
Ync9PQpDYWxsOzIwMTkuMDYuMDYgMTA6MTI6NTY7OTI1MDAzNjU40TsyNDU50Dg7Ym15c2subG9
j
Ywwuc21nLmdyLnRydW5rLmF1dG9uYW11O2JpeXNrLmxvY2FsLnNzdy5nci50cnVuay5hdXRvbmF
t
ZTswOzA7MjM7MTY7bm9ybWFsOzE7MjQ10Tg40zsxMDsxOzA70TI1MDAzNjU40Ts7Q2FsbCBpcyB
j
YW5jZWxsZWQgYnkgaW5pdG1hdG9yOzEwNDA3Mdc0NzE7Ozs7MTkyLjE2OC4yLjg7O3NtZy00O2V
t
czE7MDs7UENNQSxQQ01VLEc3MjksdGVsZXBob251LWV2ZW50OzFNcUxvRUJWVlZrdlpqdjZxRUQ
r
WEJ0eVN0N1Z2UFgrMVhVWwWyWTZJQ21tS0k1S0I0Vm5BbFJnMWNIS1NoZ2xDNDdSMm8yc1ZFb2p
J
cV1xcmRxeit3PT0KQ2FsbDsyMDE5LjA2LjA2IDEwOjEzOjExOzQ5NTc5NzkwODg7MjQ10Tg402J
p
eXNrLmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlZay5sb2NhbC5zc3cuZ3IudHJ1bms
u
YXV0b25hbWU7MDswOzEwOzE2O25vcm1hbDsxOzI0NTk40Ds7MTA7MTswOzQ5NTc5NzkwODg7O0N
h
bGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjsxMTA0MDgzMzkxOzs7OzE5Mi4xNjguMi440zt
z
bwctNDt1bXMxOzA701BDTUESUENNVsXHNzI5LHR1bGVwaG9uZS1ldmVudDtHeWRHVEtRbUlXVUJ
j
YTE3NkFQQThRZ0Eyd01HSjA5KzZZK2VktTRYbTk4Si9DM3VnS1I5ZndMVVRkVetKTVZXa0IzaXR
w
MlM3M0tDdmNLa1F2Z1I5QT09CkNhbGw7MjAxOS4wNi4wNiAxMDoyOToxNTs5MjUwMMD2NTg5OzI
0
NTk40DtiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV0b25hbWU7Ym15c2subG9jYwwuc3N3Lmd
y
LnRydW5rLmF1dG9uYW11OzA7MDsyNDsxNjtu3JtYww7MTsyNDU50Dg7OzEwOzE7MDs5MjUwMMD
2

NTg50ztDYWxsIGlzIGNhbmNlbGxlZCBieSBpbm10awF0b3I70DUyMjkxNTI30zs70zE5Mi4xNjg
 u
 Mi440ztzbwctNDt1bXMxOzA701BDTUESUENNVsXHNzI5LHR1bGVwaG9uZS1ldmVudDt0Y09lVmd
 o
 dGN5K1hVScTKdW9xewpMZURkWFVsbHpNaG5TWVJVR1BnVlBYcUN4S3NVREFnZ3FPZ0o30XRRb3p
 G
 VGw3dmY40DBHeStxSVJNMEVawUpZUT09CkNhbGw7MjAxOS4wNi4wNiAxMDoyOTozNTs5NjUwMjU
 y
 NjExOzI0NTI4NTtiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV0b25hbWU7Ym15c2subG9jYww
 u
 c3N3LmdyLnRydW5rLmF1dG9uYW1lOzA7MDsyNDsxMDI7bm9ybWFsOzE7MjQ1Mjg10zSxMDsxOzA
 7
 OTY1MDI1MjYxMTs7Q2FsbCBpcyBjYW5jZWxsZWQgYnkgaw5pdG1hdG9yOzkzNTU20Tg4NTs70zs
 x
 OTIuMTY4LjIuODs7c21nLTQ7ZW1zMTswOztQQ01BLFBDTVUsRzcyOSx0ZWxlGhvbUUtZXZlbnQ
 7
 UFFFWwXROVFsNwDdcUZrU2dTR1VyUC9oM1NacmduQ3h1S0syZmY4a3pUdHgyaStHTnVpd05NV1d
 q
 Mis0ZVE1Y25kMnBLU1oycEg3ejQvS2V2SVV4Z3c9PQpDYWxsOzIwMTkuMDYuMDYgMTA6NDI6MjE
 7
 OTIzNDg5MTUyNTsyNDU10Tc7Ym15c2subG9jYwwuc21nLmdyLnRydW5rLmF1dG9uYW1lO2JpeXN
 r
 LmxvY2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTswOzA7MjQ7MTAy025vcmlhbDsxOzI0NTU5Nzs
 7
 MTA7MTswOzkyMzQ0TE1MjU700NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjs0MTQ4NDg
 w
 NTYzOzs70zE5Mi4xNjguMi440ztzbwctNDt1bXMxOzA701BDTUESUENNVsXHNzI5LHR1bGVwaG9
 u
 ZS1ldmVudDt1T1R2S1U4VHPLQ2RFbkxVaEZFcZdocUFpRkQza21HSHdidGdHOW1zME15UmpxVW1
 t
 a2MvYkYrc1o2QVp10UYzMklvekxGZERCNWFLzhyZW0zM0lvQT09CkNhbGw7MjAxOS4wNi4wNiA
 x
 MDo0Mj00Nzs5MjM0ODkxNTI1OzI0NTU5NztiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV0b25
 h
 bWU7Ym15c2subG9jYwwuc3N3LmdyLnRydW5rLmF1dG9uYW1lOzA7MDs50zE2025vcmlhbDsxOzI
 0
 NTU5Nzs7MTA7MTswOzkyMzQ0TE1MjU700NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjs
 0
 MjU3NjM0MTgxOzs70zE5Mi4xNjguMi440ztzbwctNDt1bXMxOzA701BDTUESUENNVsXHNzI5LHR
 1
 bGVwaG9uZS1ldmVudDtwVDFWNmgxWGZ0NDhEQWpSYnVGTzJiek4rMU1Qcm9QVjBiTF1tMkw1MGd
 k
 UmJHa0ZRTVJFaHBQTH1INngyei9E0H1MaU9yLzIyTGoZMkJsN1RJY0dtQT09CkNhbGw7MjAxOS4
 w
 Ni4wNiAxMD00DoxMDS5NjIzMTMyNjQ0OzI0NTYyNjtiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bms
 u
 YXV0b25hbWU7Ym15c2subG9jYwwuc3N3LmdyLnRydW5rLmF1dG9uYW1lOzA7MDsyNDsxODthUHR
 5

RGlzYzsx0zI0NTYyNjs7MTA7MTsw0zk2MjMxMzI2NDQ700NhbGwgaXMgY2FuY2VsbGVkIGJ5IG1
u
aXRpYXRvcjsxMzE30TM00TE10zs70zE5Mi4xNjguMi440ztzbwctNDtIbXMx0zA701BDTUESUEN
N
VSxHNzI5LHR1bGVwaG9uZS1ldmVudDtZSDQ5SGgzMk12QVMYlZlZ1Yk0xTTFBRDFPyZQxZ2hNN1F
S
Z3FqZC9aNmtTNHVLcVdreGNnS0ZvUjd10E9VT01mcU1sYkZqQkpDMzZ0bTNqY0VXd3pRZz09CkN
h
bGw7MjAxOS4wNi4wNiAxMDo1MT0xNTs5MjM00DkxNTI10zI0NTA10DtiaXlZay5sb2NhbC5zbWc
u
Z3IudHJ1bmsuYXV0b25hbWU7Ym15c2subG9jYwwuc3N3LmdyLnRydW5rLmF1dG9uYW1l0zA7MDs
y
NDsxMDI7bm9ybWFs0zE7MjQ1MDU40zsxMDsx0zA70TIzNDg5MTUyNTs7Q2FsbCBpcyBjYW5jZWx
s
ZWQgYnkgaw5pdG1hdG9y0zIw0TM1NzcXmzc70zs7MTkyLjE20C4yLjg703NtZy0002VtczE7MDs
7
UENNQSxQQ01VLEc3MjksdGVsZXBob251LWV2ZW5002dtY3JKZk1wenJ3NGdJWHNuN1ZuZG10S1F
S
aVcwQ1Y5SEYyR3A2TmRheG5JZ1dZWH1MaUpMaFk1WmR5e1VCVUxWU0t2Q29yWFJ3SEx5QXBkY2F
N
ZFVRPT0KQ2FsbDsyMDE5LjA2LjA2IDew0jUx0jQz0zkyMzQ40TE1MjU7MjQ1MDU402JpeXNrLmx
v
Y2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlZay5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV0b25
h
bWU7MDsw0zc7MTY7bm9ybWFs0zE7MjQ1MDU40zsxMDsx0zA70TIzNDg5MTUyNTs7Q2FsbCBpcyB
j
YW5jZWxsZWQgYnkgaw5pdG1hdG9y0zIyMTA2ODY1MjY70zs7MTkyLjE20C4yLjg703NtZy0002V
t
czE7MDs7UENNQSxQQ01VLEc3MjksdGVsZXBob251LWV2ZW5002dPUFJSU0VFNjErZG03YWRJVVF
K
SmhzRC9HSnR5cUJqZG5KRfJZTG9mM3ZLcVUrZm1CQUVUdDRGU2txN0d0UTNMUFkvNStnWTF5UFN
F
ci9vaXhHMEtnPT0KQ2FsbDsyMDE5LjA2LjA2IDew0jU20jU50zQ5NTEzOTg10Dc7MjQ2NzY402J
p
eXNrLmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlZay5sb2NhbC5zc3cuZ3IudHJ1bms
u
YXV0b25hbWU7MDsw0zk7MjE7YVB0eURpc2M7MTsyNDY3Njg70zIy0DsX0zA7NDk1MTM50DU4Nzs
7
Q2FsbCBpcyBjYW5jZWxsZWQgYnkgaw5pdG1hdG9y0zM1MzYzE5MDQ70zs7MTkyLjE20C4yLjg
7
03NtZy0002VtczI7MDs7UENNQSxQQ01VLEc3MjksdGVsZXBob251LWV2ZW500zN0SmJ4Wld0QWU
Z
wko3aw04c0s2SV1hYkUxTThGVWI4TGdDN311NFN6eStPQwpwTDd0VVRDwFR0Si9RTjZ3eDhVKzN
M
R1ZRSFF6aDBKTH1TeDFmT1JnPT0KQ2FsbDsyMDE5LjA2LjA2IDew0jU40jI50zk2NTc4NTU40TE
7
MjQ1NDUx02JpeXNrLmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlZay5sb2NhbC5zc3c
u

Z3IudHJ1bmsuYXV0b25hbWU7MDswOzI0OzEwMjtub3JtYw7MTsyNDU0NTE7OzEwOzE7MDS5NjU
3
ODU10DkxOztDYWxsIGlzIGNhbmNlbGxlZCbieSBpbm10awF0b3I7MzKxMzM30DAzMDs7OzSxOTI
u
MTY4LjIuODs7c21nLTQ7ZW1zMTswOztQQ01BLFBDTVUsRzcyOSx0ZWxlGhVbmUtZXZlbnQ7TTN
k
eXQ5U2YxdXNwazRXQ3ltQ2lnVkQ5SjhnZHRyZEtUytodmRmNXJ5TzZEzi9yK2JiU3lCK0F3dUM
v
VGh0TXF4ZExCWw1ZRHNQaUU1enpVMEJvMlE9PQpDYWxsOzIwMTkuMDYuMDYgMTE6MDA6MTM7OTY
1
Nzg1MDk3MDSyNDU2MDA7Ym15c2subG9jYwuc21nLmdyLnRydW5rLmF1dG9uYW1lO2JpeXNrLmx
v
Y2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTswOzA7MjQ7MTAyO25vcmlhbDsxOzI0NTYwMDS7MTA
7
MTswOzk2NTc4NTA5NzA7O0NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjs1NTI00Dg3Njs
7
OzSxOTIuMTY4LjIuODs7c21nLTQ7ZW1zMTswOztQQ01BLFBDTVUsRzcyOSx0ZWxlGhVbmUtZXZ
l
bnQ7dG12NFphWHhzeCtSSWJoKzZ5SnVvQmN4LytiWjZKcTdDawY4aldQTUEwRkRLN1FsQldDN2l
W
cEFZN0poZ0xkOw5nb1UyRnFXVGRyej1RYkptL1dqeUE9PQpDYWxsOzIwMTkuMDYuMDYgMTE6MDY
6
NTA7OTY1Mzg4OTkyMTsyNDY2Nz7Ym15c2subG9jYwuc21nLmdyLnRydW5rLmF1dG9uYW1lO2J
p
eXNrLmxvY2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTswOzA7MjQ7MTAyO25vcmlhbDsxOzI0NjY
3
Mzs7MTA7MTswOzk2NTM4ODk5MjE7O0NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjsxNzI
w
NjYzODAwOzs7OzE5Mi4xNjguMi44OztzbWctNDt1bXMyOzA7O1BDTUESUENNVsXHNzI5LHR1bGV
w
aG9uZS1ldmVudDtzaUFEQ2xqZ3FrNEdocj1TTk1J2YWR3SDFHSTZCa1d0NGt6cWdPRUNrZwp1S2p
z
MVhLcS8rYk1BVkh0eUxNc0xXb01Ia3E5bVYZUWmzTWZVL2x1Wkd5Zz09CkNhbGw7MjAxOS4wNi4
w
NiAxMToyMDozMDs5NjQyNzg4OTU2OzI0NTI4NTtiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV
0
b25hbWU7Ym15c2subG9jYwuc3N3LmdyLnRydW5rLmF1dG9uYW1lOzA7MDSyNDsxMDI7bm9ybwF
S
OzE7MjQ1Mjg1OzSxMDSxOzA7OTY0Mjc4ODk1Njs7Q2FsbCBpcyBjYW5jZWxsZWQgYnkgaW5pdG1
h
dG9yOzgz2NDIxOTcyNjs7OzSxOTIuMTY4LjIuODs7c21nLTQ7ZW1zMTswOztQQ01BLFBDTVUsRzc
y
OSx0ZWxlGhVbmUtZXZlbnQ7R1Rwan1Xc3p0RisvcjgvwnY4cmdLQ3JucTNLV3lLaVlJR0ZndXh
q
dFY0MUJrRX1seHAwbi9rZz1FWHNFZFdJZ3l3V1JYNzBpZWJoOXhFK1Y4SVBYdke9PQpDYWxsOzI
w
MTkuMDYuMDYgMTE6MjA6NTg7OTYzNzIwMjE0DSyNDU5ODg7Ym15c2subG9jYwuc21nLmdyLnR
y

dw5rLmF1dG9uYW1102JpeXNrLmxvY2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTswOzA7MjQ7MTA
y
025vcm1hbDsX0zI0NTk4ODs7MTA7MTswOzk2MzcyMDIxNDg700NhbGwgaXMgY2FuY2VsbGVkIGJ
5
IGluaXRpYXRvcjs50DE40DUzMDk70zs7MTkyLjE2OC4yLjg703NtZy0002VtczE7MDS7UENNQSx
Q
Q01VLEc3MjksdGVsZXBob25lLWV2ZW50Oz14RU1YUldsVG8xVXl4NDhQVm5Q0UZRG1aUW96b3p
l
SEJBRzJHwmtGMkwrcjE0RjJrTDBmbXBGdVdaa01QNnZoQkhGNUxyc0NoR0paTwovSEk0anBnPT0
K
Q2FsbDsyMDE5LjA2LjA2IDEx0jIx0jIz0zQ5NTc5NzkW0Dg7MjQ10Tg402JpeXNrLmxvY2FsLnN
t
Zy5nci50cnVuay5hdXRvbmFtZTtiaXlZay5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV0b25hbWU7MDS
w
Ozk7MTY7bm9ybWFsOzE7MjQ10Tg40zsxMDSxOzA7NDk1Nzk30TA40Ds7Q2FsbCBpcyBjYW5jZWx
s
ZWQgYnkgaw5pdG1hdG9yOzEwODczMDEwNDE70zs7MTkyLjE2OC4yLjg703NtZy0002VtczE7MDS
7
UENNQSxQQ01VLEc3MjksdGVsZXBob25lLWV2ZW5001A5MnV0K0wvUU0zwnhaRnVFeVh0b2t1e1V
X
a1BTTF0rZ1FJRG11Y2FVdUFqZW8vM1V3Zzg4a1Q1UnpjS3VrbWlSSDZi0DdKRG1qaW1kV21oTTV
l
eXhRPT0KQ2FsbDsyMDE5LjA2LjA2IDEx0jI30jM30zkyNTAwMTk3NDQ7MjQ10Tg402JpeXNrLmx
v
Y2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlZay5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV0b25
h
bWU7MDSwOzIz0zE2025vcm1hbDsX0zI0NTk4ODs7MTA7MTswOzkyNTAwMTk3NDQ700NhbGwgaXM
g
Y2FuY2VsbGVkIGJ5IGluaXRpYXRvcjsyNjU1Njk2NTU50zs70zE5Mi4xNjguMi440ztzbWctNDt
l
bXMxOzA701BDTUESUENNVSxHNzI5LHRlbgVwaG9uZS1ldmVudDtZGZVsm3VSSk9MV1NJmzN1MDZ
y
Rk9BWXBaRnppZVFLN3ZxTG41Ti9xVTJmRlVWNlkvVmNZUCs4dEVpVzFCS0lpTDFhR3pBcTRVRz1
5
ZFZvUm10U1hwUT09CkNhbGw7MjAxOS4wNi4wNiAxMT0zMzo1MjZ0DUzMDIyMDM10zI0NTU0Njt
i
aXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV0b25hbWU7Ym15c2subG9jYwuc3N3LmdyLnRydw5
r
LmF1dG9uYW110zA7MDSyNTsX0Dtub3RSZWFjaGFibGU7MTsyNDU1NDY70zEwOzA7MDSz0DUzMDI
y
MDM10zt0ZXJtaW5hdGlvbiBzaWRlIG5vdCBYZXNwb25zZTs0MjI30Tk3NjIx0zs70zE5Mi4xNjg
u
Mi440ztzbWctNDt1bXMxOzA701BDTUESUENNVSxHNzI5LHRlbgVwaG9uZS1ldmVudDtIZ29XaE9
R
eVRGUW51Tl1k5ZHcrcXVySwhRYUxUcUE3RW1SK2dqSWJtZfQ3bEE3dmNkWEV1L3JYbmxhK3dFYmR
5
U2ZYa0RVU3RbcnNwZm40NGQ2WjRzUT09CkNhbGw7MjAxOS4wNi4wNiAxMT0zNDoxOTsz0DUzMDI
y

SzFSZHBVWmc9PQpDYWxsOzIwMTkuMDYuMDYgMTE6Mzg6MDQ7Mzg1MzAyMjAzNTsyNDU1NDY7Ym1
5
c2subG9jYwuc21nLmdyLnRydW5rLmF1dG9uYW1lO2JpeXNrLmxvY2FsLnNzdy5nci50cnVuay5
h
dXRvbmFtZTswOzA7MDsxNjtu3JtYww7MTsyNDU1NDY7OzEwOzE7MDszODUzMDIyMDM1OztDYWx
s
IGlzIGNhbmNlbGx1ZCBieSBpbm10aWF0b3I7OTkwNzU5ODI5Ozs7OzE5Mi4xNjguMi44OztzbWc
t
NDt1bXmxOzA7O1BDTUESUENNVsXHNzI5LHR1bGVwaG9uZS1ldmVudDtyaUxKw1BsekxEQ3ZoeFV
L
NkZSY2F4ZFdTmG1xVndFTmowdjQvc1hYemJnaHQ5R3VZeVZDbEN1K1F0wm1KVEQ4Q0k2ZEREQ2R
M
UUnbhmhIaTV3YmU2Zz09CkNhbGw7MjAxOS4wNi4wNiAxMT00MzoyMTs5MDM1ODA4NzUyOzI0NTQ
0
MDtiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV0b25hbWU7Ym15c2subG9jYwuc3N3LmdyLnR
y
dW5rLmF1dG9uYW1lOzA7MDsyNDszMTtub3JtYww7MTsyNDU0NDA7OzEwOzE7MDs5MDM1ODA4NzU
y
OztDYWxsIGlzIGNhbmNlbGx1ZCBieSBpbm10aWF0b3I7MjMyMDIxMzcwOTs7OzxsOTIuMTY4LjI
u
ODs7c21nLTQ7ZW1zMTswOztQQ01BLFBDTVUsRzcyOSx0ZWxlGhvbUUtZXZlbnQ7UzgzSnNXR0N
v
U1lUZTJvTFJpeVlRwmtkZ1Q3NjZEdDBHskN4T29wTW9VbWZaUDF0SwtXM09RVkZ3T2RuR1dVVDR
S
YmtrTG9PNStNaz1a0GhBNjdtR1E9PQpDYWxsOzIwMTkuMDYuMDYgMTE6NDQ6MjM7Mzg1NDI0MDQ
2
MjsyMjQwNDY2Ozs7MDswOzA7MztPbnZhbG1kTnVtYmVyOzE7MjI0MDQ2Njs7MTA7MDswOzM4NTQ
y
NDA0NjI7OyJSdWx1ICi1ZGVmYXVsdF9ub19yb3V0ZSIiIGZpbm1zaGVkIHdpdGggbm8gcm91dGU
i
OzI10DAzOTQ2Njg7Ozs7MTkyLjE2OC4yLjI2OzswNjE2ZTFjZGY2ZDE3ZW10zswOztQQ01BLEc
3
MjksRzcyMyxQQ01VLHR1bGVwaG9uZS1ldmVudDtZMDF6dnc5UEZibTNNMXJTNnFLTXRUZHF1eDF
X
aGM4Z2ZCZ1pjNG42MudJd1pUVmY4MmVSczZST0E4bXFQYThYNmxq0XdCZ1hPeGc5c1NwOUluQm9
F
UT09CkNhbGw7MjAxOS4wNi4wNiAxMT00NjoyMzs5MDk5NDUxMzA5OzI0NTk4ODtiaXlZay5sb2N
h
bC5zbWcuZ3IudHJ1bmsuYXV0b25hbWU7Ym15c2subG9jYwuc3N3LmdyLnRydW5rLmF1dG9uYW1
l
OzA7MDsyNDszMTtub3JtYww7MTsyNDU5ODg7OzEwOzE7MDs5MDk5NDUxMzA5OztDYWxsIGlzIGN
h
bmN1bGx1ZCBieSBpbm10aWF0b3I7MzA4MzI1NDkxMzs7OzxsOTIuMTY4LjIuODs7c21nLTQ7ZW1
z
MTswOztQQ01BLFBDTVUsRzcyOSx0ZWxlGhvbUUtZXZlbnQ7R3BjTlVsMkZTd0pZdWREWwpsN3V
K
VUtjS1dxNTJ1dUsySTUzbk1Ywml3a1B6d01xenRNSUxsVVJBNjAvRXpIRkdBwnBCNjNzdEdVaEg
0

QmZGbFR3Y2c9PQpDYWxsOzIwMTkuMDYuMDYgMTE6NTI6NDI70TA5NzI3NzcwODsyNDUyODU7YmI
5
c2subG9jYwuc21nLmdyLnRydW5rLmF1dG9uYW1lO2JpeXNrLmxvY2FsLnNzdy5nci50cnVuay5
h
dXRvbmFtZTswOzA7MjQ7MTAyO25vcmlhbDsxOzI0NTI4NTs7MTA7MTswOzkwOTcyNzc3MDg700N
h
bGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjszNzgxNzUyMdc7Ozs7MTkyLjE2OC4yLjg703N
t
Zy00O2VtczE7MDS7UENNQSxQQ01VLEc3MjksdGVsZXBob25lLWV2ZW5001RrYWtxaEhycFZQT25
q
Z2t0eXp2di8yUDRWUk16cG5mRTRncDVQNjk3bnNhbFNha1BUSmZDQTI1QjFPQnV4ZndqUEZYQUk
w
OGhaY1h0ckNzVFB5NmFRPT0KQ2FsbDsyMDE5LjA2LjA2IDExOjUzOjE0OzM4NTQyNTUwMTQ7Kjc
z
IztiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV0b25hbWU7OzEyNTswOzA7MTI303N5c3R1bUZ
h
aWx1cmU7NzsyNDA10Tg7OzEwOzA7MDSzODU0MjU1MDE0OztSZWx1YXNlIGZyb20gTVNSOiBtc3J
f
bm90X2F2YWlsYWJsZTs1MTI1NTAyMDE7MjQwNTk4Oyo3MyM7Mzg1NDQxNTc10TsxOTIuMTY4LjI
u
ODs7c21nLTQ7MDYxNmUxY2UwNDNhZmU4NTsxOztUMzg7bXh2dmFlM1dSMGVkQ3Roc0RVMWdxc1V
j
bW50azJ2WGo3aHNmDHV5MlZVVXVqbDhUd3BNcu16N0c5Qys1QnVrT291d1VaT255cURERDdBaE9
i
Ylp1Unc9PQpTZXJ2aWNlOzIwMTkuMDYuMDYgMTE6NTM6MTQ7MjQwNTk4Oyo3MyM7OzswOzA7MDS
w
Ozs702ludm9jYXRpb247OzsxOzI0MDU5ODtjZnU7OzUxMjU1MDIwMTszODU0MjU1MDE0OzsyNDA
1
OTg7Ozs7Ozs700xya0ZjMmdTSFJBNGhUZVVZeGtuWUpUSnNk0VxkZWczejJkR2RZeTgxZG5oY0V
W
L3BCd2NqSkTCZWo2MElQqmlRUFNTYWh5YXl2RDJXY0x5UjQwQVhRPT0KQ2FsbDsyMDE5LjA2LjA
2
IDExOjU2OjA5OzM4NTQyNTUwMTQ7KjczIztiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV0b25
h
bWU7OzEyNTswOzA7MzE7bm9ybWFsOzc7MjQwNTk4OzswMDswOzA7Mzg1NDI1NTAxNDs7VGvybWl
u
YXRlZCBieSBzeXN0ZW0gYWRTaw5pc3RyYXRvcjsxMjQ10DE10DA4OzI0MDU5ODsqNzMjOzM4NTQ
0
MTU3NTk7MTkyLjE2OC4yLjg703NtZy00OzA2MTZlMWNlMDQzYWZlODU7MTs7VDM401lmb3VSSXp
Q
Q3FQMwVyZnhid2UvdHJDY1hEbwDyeFJiaVZCL1A0S1pmZ3lZclowTwhxZVRIIdDFVTTNiL2w2M3J
S
ejhGUjZPeXEvCVjpbFRZajdGbFlNPT0KU2Vydm1jZTsyMDE5LjA2LjA2IDExOjU2OjA5OzI0MDU
5
ODsqNzMjOzs7MDswOzA7MDS7OztpbnZvY2F0aw9uOzs7MTsyNDA10Tg7Y2Z1OzswMjQ10DE10DA
4
OzM4NTQyNTUwMTQ7OzI0MDU5ODs7Ozs7Ozs7RnoyZkkrdGs0TmJuQnZXUz1Pb3BqL3J0Si9HZDl
P

WUtad0lPbzh5eEVyT0JzYmRzVWR2Y3hKMlZORHRCV2o0dTlFZXVCa2l6RUxUVTgxaWpjMXlhr1E
9
PQpDYWxsOzIwMTkuMDYuMDYgMTE6NTY6NDE70TA5NjI5NDQyMDSyNDU0NDA7Ym15c2subG9jYWw
u
c21nLmdyLnRydW5rLmF1dG9uYW1l02JpeXNrLmxvY2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTs
w
OzA7MjQ7MzE7bm9ybWFsOzE7MjQ1NDQwOzxsMDsxOzA70TA5NjI5NDQyMDS7Q2FsbCBpcyBjYW5
j
ZWxsZWQgYnkgaw5pdG1hdG9yOzEzODA20Tk3MjU70zs7MTkyLjE20C4yLjg703NtZy0002VtczE
7
MDs7UENNQSxQQ01VLEc3MjksdGVsZXBob25lLWV2ZW50005XVjlvvcitvekxHUGFzd1htYkxoUGh
z
b1NsQTJyazEwMC95OUhXMnpBL0VHempwbnNubUJwd3JabjY0Nk9SUFBMSmxGw1MwZHlXcGNBTEX
5
SWZaM1FBPT0KQ2FsbDsyMDE5LjA2LjA2IDExOjU50jAwOzk2NTM4ODUzMDI7MjQ2NTcxO2JpeXN
r
LmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlZay5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV
0
b25hbWU7MDswOzI0OzMxO25vcm1hbDsxOzI0NjU3MTs7MTA7MTswOzk2NTM4ODUzMDI700NhbGw
g
aXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjsxOTYzMzIxMjMxOzA70TA5NjI5NDQyMDSyNDU0NDA7
t
NDt1bXMyOzA701BDTUESUENNVSxHNzI5LHRlbGVwaG9uZS1ldmVudDtpdjNleEZwU0x0Vk1KaEd
l
UXg5N0QxZkRrVUk4NUF0aVRmUTZaeVlPc0p2NHdaKy90endHeDJaZ3c4THdFOS9xcmZPZGcyVHB
j
MDE5THhmd0UrL2xGdz09CkNhbGw7MjAxOS4wNi4wNiAxMTo10TowOTszODU0MjU1MDE00yo3MyM
7
Ym15c2subG9jYWwuc21nLmdyLnRydW5rLmF1dG9uYW1l0zxsMjM4ODUzMDswOzE7MjQ1NDQwOz
s
dXJl0zc7MjQwNTk4OzxsMDswOzA7Mzg1NDI1NTAxNDs7UmVsZWZzZSBmcm9tIE1TUjogbXNyX25
v
dF9hdmFpbGFibGU7MjAwMTA2NzA4NjA4NjA4NjA4NjA4NjA4NjA4NjA4NjA4NjA4NjA4NjA4NjA4
4
OztzbWctNDswNjE2ZTFjZTA0M2FmZTg10zE701Qz0DtW0EtKaDJNT1Z0YmRvK21xczI1b1N4eCs
v
Wwt0ZUhXWNCTUxKa281a2R6ekhWbnQydfN4ZE9Ud2FCUWNCaEthZzhi0HFJWHRueUw3ejFoaUd
3
NXhtUT09C1N1cnZpY2U7MjAxOS4wNi4wNiAxMTo10TowOTsyNDA10Tg7KjczIzs70zA7MDswOzA
7
Ozs7aW52b2NhdG1vbjs70zE7MjQwNTk4O2NmdTs7MjAwMTA2NzA4NjA4NjA4NjA4NjA4NjA4NjA4
1
OTg70zs70zs702xPdGYwOGhuUnVMakxYbXhqRFVTWdNaMzhISlBGSEJLdm1lVwo0bWp0Q1dEN0Z
q
M1I1bmdzSTBPUUNBUmdEUMVpT3dzUytSazR0cFRyQWtvNU13V0pRPT0KQ2FsbDsyMDE5LjA2LjA
2
IDEyOjIxOjU5Ozk2Mdc5MzE1Mjc7MjQ2NTcxO2JpeXNrLmxvY2FsLnNtZy5nci50cnVuay5hdXR
v

bmFtZTtiaXlZay5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV0b25hbWU7MDswOzI0OzEwMjtub3JtYWw7
7
MTsyNDY1NzE7OzEwOzE7MDs5NjA3OTMxNTI3OztDYWxsIGlzIGNhbmNlbGxlZCBieSBpbml0aWF0
0
b3I7MzQ1MTk3Mzc2Mjs7OzxsOTIuMTY4LjIuODs7c21nLTQ7ZW1zMjswOztQQ01BLFBDTVUsRzcy
y
OSx0ZWxlGhvbmUtZXZlbnQ7aGJWMS82OUJ1Qkwycm9SaEntYXpacUQzOUgrT3NrOD1EU21YZWdh
h
a0NkVV1YbHlCUMh6UDJFQTRiSVRBWw1uaUptT2hyRHVuT1BveHZLeVpMwk1GZwc9PqpDYWxsOzI
w
MTkuMDYuMDYgMTI6Mjg6NDE7OTI2NDUwNzYxODsyNDY2Nz7Ym15c2subG9jYwwuc21nLmdyLnR
y
dw5rLmF1dG9uYW11O2JpeXNrLmxvY2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTswOzA7MjQ7MTg
7
YVB0eURpc2M7MTsyNDY2Nz7OzEwOzE7MDs5MjY0NTA3NjE4OztDYWxsIGlzIGNhbmNlbGxlZCB
i
eSBpbml0aWF0b3I7ODQzMzU2MjEwOz7OzE5Mi4xNjguMi44OztzbWctNDt1bXMyOzA701BDTUE
s
UENNVsXHNzI5LHR1bGVwaG9uZS1ldmVudDtjTnQ1MzFkeEdmbm5XSmM2SWRYczMyc1QvVV1mbHV
v
M214cE9mNTFGQ2FIYmxyZUZpbk5tYzAzaGEwa2djTzFraXhJwko3a0V30EF5NStQYTNTcVJpZz0
9
CkNhbGw7MjAxOS4wNi4wNiAxMjoyOToyMDs5NjQ4MzU0NjE4OzI0NjU3MTtiaXlZay5sb2NhbC5
z
bWcuZ3IudHJ1bmsuYXV0b25hbWU7Ym15c2subG9jYwwuc3N3LmdyLnRydW5rLmF1dG9uYW11OzA
7
MDsyNDsxMDI7bm9ybWFsOzE7MjQ2NTcxOzxsMDsxOzA70TY00DM1NDYxODs7Q2FsbCBpcyBjYW5
j
ZWxsZWQgYnkgaw5pdG1hdG9yOzEwMDczMjk5Mzc7Oz7MTkyLjE2OC4yLjg703NtZy0002VtczI
7
MDs7UENNSXQ01VLEc3MjksdGVsZXBob251LWV2ZW5002FTZnZ0YjhJbkZ0ZVBvRFZVWVpw0VA
y
ZkZnQy9kNzJjOVixSVp5cmUyWWRURnVkaThCMzNPe1FsV0RkMDBJVUk5M28wei9jZD16M09YbnV
0
YkpqOGJBPT0KQ2FsbDsyMDE5LjA2LjA2IDEyOjM4OjA1OzkyNTAwMTk3NDQ7MjQ1NjI2O2JpeXN
r
LmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlZay5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV
0
b25hbWU7MDswOzI0OzE2O25vcm1hbDsxOzI0NTYyNjs7MTA7MTswOzkyNTAwMTk3NDQ700NhbGw
g
aXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjszMjA4OTQ4NzEwOz7OzE5Mi4xNjguMi44OztzbWc
t
NDt1bXMyOzA701BDTUESUENNVsXHNzI5LHR1bGVwaG9uZS1ldmVudDtEMU9KaUJLb0I2QkI2T3Z
4
N1dCeZFUitlUmVxY0VEN3hLdmRDTDR0c2tnbFMvNityaTdiV21TRzZFd1A5RWZIN2dv0GpodlF
v
d21MK21RRVd1NDJqQT09CkNhbGw7MjAxOS4wNi4wNiAxMj0zOTowNz5MjY0NTA3NjE4OzI0NjY
3

MztiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV0b25hbWU7Ym15c2subG9jYWwuc3N3LmdyLnRydw5rLmF1dG9uYW1lOzA7MDsyNDsxODthUHR5RG1zYzSxOzI0NjY3Mzs7MTA7MTsw0zkyNjQ1MDc2MTg700NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjszNDY5MjQ5NTczOzs70zE5Mi4xNjguMi440ztzbWctNDt1bXMyOzA701BDTUESUENNVsXHNzI5LHR1bGVwaG9uZS1ldmVudDtGe1E5cmdpbU14VnhhdXhTaGJBQlgwZk43ZndGMGV5NzN6Y3JldzU3TkY0TlNlIbGtRwmlINUM2M1l0QTl0Y3k5VU82UStpYzdqY1E0R1lGWGVs0Xcydz09CkNhbGw7MjAx0S4wNi4wNiAxMj00ND0zNjs5NjIzMTQwODUxOzI0NjY3MztiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV0b25hbWU7Ym15c2subG9jYWwuc3N3LmdyLnRydW5rLmF1dG9uYW1lOzA7MDsyNDsxMDI7bm9ybWFsOzE7MjQ2NjczOzsxMDsxOzA70TYyMzE0MDg1MTs7Q2FsbCBpcyBjYW5jZWxsZWQgYnkgaw5pdGlhdG9yOzU1NDUyODA1Mjs70zsxOTIuMTY4LjIuODs7c21nLTQ7ZW1zMjswOztQQ01BLFBDTVUsRzcyOSx0ZWxlclGhvbmlUzXZlbnQ7SFN5b2MrRlJ4VFdyUGQ5c0xwMw9aa2VITitZazJ2ewh6dzNUcm9mNXZCUVhKQzlaemJPVkJ0NW9ZZm4c3drR3Z1Q1Y2amNXeVRrM1NHaxJoZ1Vld0E9PQpDYWxsOzIwMTkuMDYuMDYgMTI6NDU6MDA7OTIyNDA0NzM5MjsyNDYzNTM7Ym15c2subG9jYWwuc21nLmdyLnRydW5rLmF1dG9uYW1lO2JpeXNrLmxvY2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTswOzA7MjQ7MTAy025vcm1hbDsxOzI0NjM1Mzs7MTA7MTsw0zkyMjQwNDczOTI700NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjs2NTQ2NzQ1NDY70zs7MTkyLjE2OC4yLjg703NtZy0002VtczI7MDs7UENNSxQQ01VLEc3MjksdGVsZXBob25lLWw2ZW5002JYeXhhOHERd3hkc2s2MXluwVJMvkFuQn1l0UtKcXJIZmFab2xHRZlWMDZ3S1pqUUZNaI9vdFBHUTFWa1p1a3dhSTBwNVhqQTRPUkhHS1U2bWk5UXZRPT0KQ2FsbDsyMDE5LjA2LjA2IDEyOjQ10jI20zkw0TY4MTA2Nzk7MjQ1NzA402JpeXNrLmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlZay5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV0b25hbWU7MDswOzI0OzMx025vcm1hbDsxOzI0NTcwODs7MTA7MTsw0zkw0TY4MTA2Nzk700NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjs3NjM00TQyMjU70zs7MTkyLjE2OC4yLjg703NtZy0002VtczE7MDs7UENNSxQQ01VLEc3MjksdGVsZXBob25lLWw2ZW5001R1U1RtT2hpV0laRG1TZmhYRmp0M0o4UG1KTxVZRwZXU0pnQVF4NjR0enhawEZYbXJoYzkyZTBra1Q30WRWFFEU3NJZn1Kwk9iRUFmYnY0TGRRT2JnPT0KQ2FsbDsyMDE5LjA2

LjA2IDEy0jQ20jUx0zM4NTQzNDQwMzI7MjQ1Njcz02JpeXNrLmxvY2FsLnNtZy5nci50cnVuay5h
dXRvbmFtZTtiaXlzaY5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV0b25hbWU7MDsw0zI10zE4025vdFJ
l
YwNoYWJsZTsx0zI0NTY3Mzs7MTA7MDsw0zM4NTQzNDQwMzI703Rlcm1pbmF0aw9uIHNPZGUgYm9
0
IHJlc3BvbnNl0zExMjAyNzU4NTM70zs7MTkyLjE20C4yLjg703NtZy0002VtczE7MDs7UENNQSx
Q
Q01VLEc3MjksdGVsZXBob25lLWV2ZW5000hiL3EwU1UrUnc5L0dKeFlsb2x3RE1tQ1Fo0Hhch1h
P
c3k1RmMxMGJBvitveDJ2b2dpQm1VM1JiSwphdWl3SlZTM2htS0oyeTh4azRPM1oyS3dLT1V3PT0
K
Q2FsbDsyMDE5LjA2LjA2IDEy0jQ30jE50zM4NTQzNDQwMzI7MjQ1Njcz02JpeXNrLmxvY2FsLnN
t
Zy5nci50cnVuay5hdXRvbmFtZTtiaXlzaY5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV0b25hbWU7MDs
w
0zM7MTY7bm9ybWFs0zE7MjQ1Njcz0zsxMDsx0zA7Mzg1NDM0NDAAzMjs7Q2FsbCBpcyBjYW5jZWx
s
ZWQgYnkgaw5pdG1hdG9y0zEyMzc2MjU5MDA70zs7MTkyLjE20C4yLjg703NtZy0002VtczE7MDs
7
UENNQSxQQ01VLEc3MjksdGVsZXBob25lLWV2ZW5003hMdmIRUFowXdkA0NoTW8zemUrUzk3cUN
Z
ZjJtRW10cnRswsvVn1XTGsxclFNSEV2L1VvMnpTaGZyd1UwU2RPSTNhTct3ZXBzT1NBTUQzd1Y
r
0FhBPT0KQ2FsbDsyMDE5LjA2LjA2IDEy0jQ30jI20zM4NTQzNDQwMzI7MjQ1Njcz02JpeXNrLmx
v
Y2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlzaY5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV0b25
h
bWU7MDsw0zI10zE4025vdFJlYWNoYWJsZTsx0zI0NTY3Mzs7MTA7MDsw0zM4NTQzNDQwMzI703R
l
cm1pbmF0aw9uIHNPZGUgYm90IHJlc3BvbnNl0zEYNjc1NDk2NjQ70zs7MTkyLjE20C4yLjg703N
t
Zy0002VtczE7MDs7UENNQSxQQ01VLEc3MjksdGVsZXBob25lLWV2ZW5001Z6TEMwaE9EVTMyeXd
h
Wms0NDFLTE1Pa2wrbUNtcUtjNVN6TC9Qc2szNHVQZjdsRnB5L3FFRT1UU042Vmcwc1k5czNZZnF
x
SXk2eVJiRk5lSFZXRTNBPT0KQ2FsbDsyMDE5LjA2LjA2IDEy0jQ30jU00zM4NTQzNDQwMzI7MjQ
1
Njcz02JpeXNrLmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlzaY5sb2NhbC5zc3cuZ3I
u
dHJ1bmsuYXV0b25hbWU7MDsw0zA7MTY7bm9ybWFs0zE7MjQ1Njcz0zsxMDsx0zA7Mzg1NDM0NDAA
Z
Mjs7Q2FsbCBpcyBjYW5jZWxsZWQgYnkgaw5pdG1hdG9y0zEzODQ4MDAxMTQ70zs7MTkyLjE20C4
y
Ljg703NtZy0002VtczE7MDs7UENNQSxQQ01VLEc3MjksdGVsZXBob25lLWV2ZW5003hUVk9WMDR
H
ejJMei8rN2lvTHN2UUErew05ZU5KSk1CwnJoVkpdsU5vRj1YRGRZNEwrOXVWUzliV3M4a2RubHF
U

V1NEeD12NEZkNTNCUXZhT1VCWTB3PT0KQ2FsbDsyMDE5LjA2LjA2IDEy0jQ40jQ20zkyMjQwNDc
Z
0TI7MjQ2MzUz02JpeXNrLmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlzaY5sb2Nhbc5
Z
c3cuZ3IudHJ1bmsuYXV0b25hbWU7MDswOzI0OzEwMj tub3JtYww7MTsyNDYzNTM70zEwOzE7MDs
5
MjI0MDQ3MzkyOztDYWxsIGlzIGNhbmNlbGxlZCBieSBpbm10aWF0b3I7MTYwMjYwNzE3MDs70zs
x
0TIuMTY4LjIuODs7c21nLTQ7ZW1zMjswOztQQ01BLFBDTVUsRzcyOSx0ZWxlGhvbUUtZXZlbnQ
7
V3kyaU02VzJZWFRFTnRkc29rU2JrTERyc0NwM1NHR204T2h0aFAyc3Rxv0ZqU3pCUVdPUWRZeVB
2
Y0t1Sz1JT251VzJiUmJHOGFnU25YT0pjamwwZHc9PQpDYWxsOzIwMTkuMDYuMDYgMTI6NDk6NDU
7
0TI2NDUwNzYxODsyNDY2Nz7Ym15c2subG9jYwwuc21nLmdyLnRydW5rLmF1dG9uYW1102JpeXN
r
LmxvY2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTswOzA7MjQ7MTg7YVB0eURpc2M7MTsyNDY2Nz7
7
OzEwOzE7MDs5MjY0NTA3NjE4OztDYWxsIGlzIGNhbmNlbGxlZCBieSBpbm10aWF0b3I7MTg1MDQ
x
MTIzODs70zsx0TIuMTY4LjIuODs7c21nLTQ7ZW1zMjswOztQQ01BLFBDTVUsRzcyOSx0ZWxlGh
v
bmUtZXZlbnQ7K1pxcG40c01HUTNDQ1A4NmZNRlNTaURRTDlCbXNQV1dwbDRSeGE5S1JqM21VSUp
J
TFNLTFdZS2ZyLzlid2RLSkFFazRUdzdvWm1aaHdjNVZCaGY4Q1E9PQpDYWxsOzIwMTkuMDYuMDY
g
MTI6NTc6MDU70TY1Mzg4NDc1NzsyNDY1NzE7Ym15c2subG9jYwwuc21nLmdyLnRydW5rLmF1dG9
u
YW1102JpeXNrLmxvY2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTswOzA7MjQ7MTAy025vcmlhbDs
x
OzI0NjU3MTs7MTA7MTswOzk2NTM4ODQ3NTc700NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXR
v
cjszNjk1MjEwNzIwOzs7OzE5Mi4xNjguMi44OztzbWctNDt1bXMyOzA701BDTUESUENNVsXHNzI
5
LHR1bGVwaG9uZS1ldmVudDthUUs3L09MVEJnRkpwNEE2cHhZakhvMFh1UDh6YThBcXUyd0ZoOU9
t
N01Mwi9mcnZyZkpjUmQ5SGhnNwppbTFDei9PbGtLUjhWMndoQ1diZk1vVWk1QT09CkNhbGw7MjA
x
OS4wNi4wNiAxMzowMDoyODs5MjY0NTA3NjE4OzI0NjY3MztiaXlzaY5sb2Nhbc5zbWcuZ3IudHJ
1
bmsuYXV0b25hbWU7Ym15c2subG9jYwwuc3N3LmdyLnRydW5rLmF1dG9uYW110zA7MDsyNDsxODt
h
UHR5RG1zYzsxOzI0NjY3Mzs7MTA7MTswOzkyNjQ1MDC2MTg700NhbGwgaXMgY2FuY2VsbGVkIGJ
5
IGluaXRpYXRvcjsyNTIxMzA2NzY70zs7MTkyLjE2OC4yLjg703NtZy0002VtczI7MDs7UENNSX
Q
Q01VLEc3MjksdGVsZXBob25lLWV2ZW50Ozk1MmpNYUd0MUR2bmVpb1NzODQrNW14a0NnZVdzcUk
3

ZHJacXRVM1SUKt3WVFQSkxGbwYxLzZhZ3pYUGhPUDD4S1Ew0HNOQ1RNZHNxVmV5YitUa2dnPT0
K
Q2FsbDsyMDE5LjA2LjA2IDEzOjA00jQ30zM4NTQzNTY3MDk7MjQwNTE102JpeXNrLmxvY2FsLnN
t
Zy5nci50cnVuay5hdXRvbmFtZTs7NDQ7MTswOzE2025vcm1hbDs30zI0MDUxNTs7MTA7MTswOzM
4
NTQzNTY3MDk7001TVVAgY2FsbCBpcyBjb21wbGV0ZTsxMzM4NzY1NzM4Ozs7OzE5Mi4xNjguMi4
4
OzE5Mi4xNjguMi4yNjtzZwctNDswNjE2ZTFjZTA0Mzg2OGVhOzE701BDTUESdGVsZXBob251LWV
2
ZW5000V6UWVHaEMxbDFiRnFabTMrdw92ZmZXS1Bhc2RPazJybTQ2Qm4vdm5yWVpns3N3V0VkkZn
4
UC9uYVVMmJRbkZY0WZ2bEJFV2E2RFNvWxp6QzRYSjlnPT0KQ2FsbDsyMDE5LjA2LjA2IDEzOjA
1
OjE0OzkwNTU3NDgwNzQ7MjQ1NDQwO2JpeXNrLmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTt
i
aXlZay5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV0b25hbWU7MDswOzI0Ozmx025vcm1hbDsxOzI0NTQ
0
MDs7MTA7MTswOzkwNTU3NDgwNzQ700NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjsxNDU
x
NzEzMjM2Ozs7OzE5Mi4xNjguMi44OztZwctNDt1bXMxOzA701BDTUESUENNVsXHNzI5LHR1bGV
w
aG9uZS11dmVudDtHTzhnUS9xaDNHS2xZL3YyWkFqdURnb09ndVZGcTZIVmxIbW9vbG1KT11XSDZ
r
aGVsL0ozQ1ZtT0duVHRkRmNyWTJKcVJYdC9yejJZV1VrM3R1Y2NzZz09CkNhbGw7MjAxOS4wNi4
w
NiAxMzowNT00MzszODU0MzU2NzA5OzI0MDUxNTtiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV
0
b25hbWU7OzM3OzE7MDsxNjtu3JtYw7NzsyNDA1MTU7OzEwOzE7MDszODU0MzU2NzA5OztJU1V
Q
IGNhbGwgaXMgY29tcGxldGU7MTU3MzQ10TE0NTs7Ozsx0TIuMTY4LjIuOdsx0TIuMTY4LjIuMjY
7
c21nLTQ7MDYxNmUxY2UwNDM4Njh1YTsxOztQQ01BLHR1bGVwaG9uZS11dmVudDt0MnJIN2dIQnJ
p
WmdSSXZtZnJlQ0xzenVteWV3S1pVM1c4MHAwbzJROXpm0WhEbzlMSTI0TXpaa2VNbXNhQSt5Q09
3
MjNoT3hVSVhtbVRwR31DK0hVQT09CkNhbGw7MjAxOS4wNi4wNiAxMzoxMT01NTszODU0MzQ5MzA
1
OzI0NTgzOTtiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV0b25hbWU7Ym15c2subG9jYWwuc3N
3
LmdyLnRydW5rLmF1dG9uYW11OzA7MDsyNTsxOdtub3RSZWFjaGFibGU7MTsyNDU4Mzk7OzEwOzA
7
MDszODU0MzQ5MzA1Ozt0ZXJtaW5hdGlvbiBzaWR1IG5vdCBYZXNwb25zZTs7MTMzMzExODcyOzs
7
OzE5Mi4xNjguMi44OztZwctNDt1bXMxOzA701BDTUESUENNVsXHNzI5LHR1bGVwaG9uZS11dmV
u
dDtPvFg2amN4YkxhU01MbXpyUmszV1BYZk1FSHFTWTg5TGnLT1J4cFQ0bjkwdXE2RU0zS2tNZ09
H

cmZQUdVJ0XdaWfN1c2ZrMEYxVwZ6wLzQOGQ2aXVPZz09CkNhbGw7MjAx0S4wNi4wNiAxMzoxMjox
y
Mzs00TU5MDEwOTg0zI0NTgz0TtiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV0b25hbWU7Ym1
5
c2subG9jYwuc3N3LmdyLnRydW5rLmF1dG9uYW1l0zA7Mds30zE2025vcm1hbDsx0zI0NTgz0Ts
7
MTA7MTsw0zQ5NTkwMTA50DQ700NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjszMjUxNTY
y
MDAx0zs70zE5Mi4xNjguMi440ztzbwctNDtlbXmX0zA701BDTUESUENNVsXHNzI5LHRlbGVwaG9
u
ZS1ldmVudDs5b1FwMXM5dVdTcEdws1ZpRTJ6UzJkTmZFRFY3RDFqeCt1eTd4Mk5SU0kzeGtKbWx
Z
awYrTEJ0bEk5STdqRkhYOCsra1FX0GNHNDfqcFZ3SktyeStLZz09CkNhbGw7MjAx0S4wNi4wNiA
x
MzoxMjox00Ts5MDk50DIwNjE0zI0NTcw0DtiaXlZay5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV0b25
h
bWU7Ym15c2subG9jYwuc3N3LmdyLnRydW5rLmF1dG9uYW1l0zA7MdsyNDszMTtub3JtYwW7MTs
y
NDU3MDg70zEw0zE7Mds5MDk50DIwNjE0ztdYwxsIGlzIGNhbmNlbGxlZCBieSBpbm10awF0b3I
7
MzM10Tk1NTQ1Nzs70zxs0TIuMTY4LjIu0Ds7c21nLTQ7ZW1zMTsw0ztQQ01BLFBDTVUsRzcy0Sx
0
ZWxlGhvbmUtZXZlbnQ7RmhIL2ZYT0x0STJrUzF3WgtsM1duM2xD0FhHam5XZzF6TU5iZHh0Ymx
W
UEo3dTF3QjhXQ3RVVm96dER00VVxV0phSUXqMkJycTFKbVZMSmLoc1h3ZkE9PQpDYwxs0zIwMTk
u
MDYuMDYgMTM6MTY6MjE7NDk1Nz5MjAwNTsyNDYwMTg7Ym15c2subG9jYwuc21nLmdyLnRydW5
r
LmF1dG9uYW1l0zJpeXNrLmxvY2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTsw0zA7MjQ7MTAy025
v
cm1hbDsx0zI0NjAx0Ds7MjI10zE7Mds00TU3MzkyMDA10zttdYwxsIGlzIGNhbmNlbGxlZCBieSB
p
bm10awF0b3I7NDI00Dg0Mjg3Mzs70zxs0TIuMTY4LjIu0Ds7c21nLTQ7ZW1zMjsw0ztQQ01BLFB
D
TVUsRzcy0Sx0ZWxlGhvbmUtZXZlbnQ7TEs5Vjg0Vmh4RXVTbUgzRTJxVUFwaVljUVhrcHZIaG1
0
aTk2MzJPOUFxSEM00E0wd1B3NTdQdGNvb2NnMnNpZnp3QXBPN0ZEZXZRkNuaCtkMnJrQwc9PQp
D
Ywxs0zIwMTkuMDYuMDYgMTM6MTY6NDc7NDk10TgxMTE4MTsyNDYwMTg7Ym15c2subG9jYwuc21
n
LmdyLnRydW5rLmF1dG9uYW1l0zJpeXNrLmxvY2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTsw0zA
7
MTk7Mzg7c3M3RmFpbHVyZTs0zI0NjAx0Ds7MjI40zE7Mds00TU50DExMTgx0zttdYwxsIGlzIGN
h
bmNlbGxlZCBieSBpbm10awF0b3I7NjI5Mjk4MTY70zs7MTkyLjE20C4yLjg703NtZy0002VtczI
7
Mds7UENNQsXQQ01VLEc3MjksdGVsZXBob25lLWV2ZW5003pFdG9VewhaS09TQz1LZEhxZzk3TDB
v

bWMxcHh1Y3gzK09EdjRLMUQydjFkYVJGczZTdGF3VCszQjZiTENBK00wU0Zzam1WT2ZCTkdFemp
Z
0Hp1d1p3PT0KQ2FsbDsyMDE5LjA2LjA2IDEz0jE30jA30zQ5NTk4MTEExODE7MjQ2MDE4O2JpeXN
r
LmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlzaY5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV
0
b25hbWU7MDswOzk7MTY7bm9ybWFsOzE7MjQ2MDE4OzsyMjg7MTswOzQ5NTk4MTEExODE7O0NhbGw
g
aXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjsxNDcwMjg2MzI7Ozs7MTkyLjE2OC4yLjg7O3NtZy0
0
02VtczI7MDs7UENNQsXQQ01VLEc3MjksdGVsZXBob25lLWV2ZW5002RtdVBCaE1NRWFqVnRXWVZ
4
ZkJtamRmYUJRMjZmU0FCbmF3Z3BIKy9WQkNzQnJZVHdGSDh4RnNFWlh2b1FQSlExaVZVY0M3M2h
D
RmRsSC8wbGN6ZVZBPT0KQ2FsbDsyMDE5LjA2LjA2IDEz0jIy0jM0OzQ5NTc4ODc5MDA7MjQ1MjY
2
02JpeXNrLmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlzaY5sb2NhbC5zc3cuZ3IudHJ
1
bmsuYXV0b25hbWU7MDswOzI1OzE4O25vdFJlYWNoYWJsZTsxOzI0NTI2Njs7MTA7MDswOzQ5NTc
4
ODc5MDA7O3Rlcm1pbmF0aw9uIHNPZGUgYm90IHJlc3BvbnNlOzE1MTg4MTEExMzE7Ozs7MTkyLjE
2
OC4yLjg7O3NtZy0002VtczE7MDs7UENNQsXQQ01VLEc3MjksdGVsZXBob25lLWV2ZW5003JIbWZ
F
TW4rVGJ3cE1NWUZ1MjVjRz1SeHk4VHNUa01qT2ExewZVZ3JjQjFMbnRUNXZy0XNFe1VaYXJMRnc
v
bVNGdStqaHducHl6enI3ZFB4N2NMNnNnPT0KQ2FsbDsyMDE5LjA2LjA2IDEz0jIz0jAz0zkyNTA
w
MTk3NDQ7MjQ1NDQwO2JpeXNrLmxvY2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlzaY5sb2N
h
bC5zc3cuZ3IudHJ1bmsuYXV0b25hbWU7MDswOzI0OzE2O25vcmlhbDsxOzI0NTQ0MDs7MTA7MTs
w
OzkyNTAwMTk3NDQ7O0NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXRvcjsxNjQwMTM0NjY4Ozs
7
OzE5Mi4xNjguMi44OztzbWctNDt1bXMxOzA7O1BDTUESUENNVsXHNzI5LHRlbGVwaG9uZS1ldmV
u
dDtBVnhFZzdUMwtzQkF5YUMxM1BMODB1Z0JESkZiZEVVZ05MdmY5enNqbk56VzR0Mmp6MmY5NFp
r
Qj1Wa3FVTFN2S0FYVWzqREUrZXpxa1RvKzh2SzVSdz09CkNhbGw7MjAxOS4wNi4wNiAxMzoyNjO
1
MDs5NjI5MTI1NjA2OzI0NTcwODtiaXlzaY5sb2NhbC5zbWcuZ3IudHJ1bmsuYXV0b25hbWU7Yml
5
c2subG9jYwuc3N3LmdyLnRydw5rLmF1dG9uYW1lOzA7MDsyNDszMTtub3JtYw7MTsyNDU3MDg
7
OzEwOzE7MDs5NjI5MTI1NjA2OztDYwxsIGlzIGNhbmNlbGxlZCBieSBpbml0aWF0b3I7MjU5Mjg
y
ODU5MDs7OzSxOTIuMTY4LjIuODs7c21nLTQ7ZW1zMTswOztQQ01BLFBDTVUsRzcyOSx0ZWxlGh
v

bmUtZXZ1bnQ7e1l4YVJTRXVmcwVmNE94NFZUC0ZibHZUMjVOL0ZWTTVvNzc0awZ2c1l0UEJlZFh
F
V0xKZ3lIRjdZNG50TlBqOU1zekNwUUZCdUJBK1pwR0VmcwJuQ3c9PQpDYWxs0zIwMTkuMDYuMDY
g
MTM6Mjk6MTY70TIzNDg5MTUyNTsyNDY1NjE7Ym15c2subG9jYwwuc21nLmdyLnRydW5rLmF1dG9
u
YW1l02JpeXNrLmxvY2FsLnNzdy5nci50cnVuay5hdXRvbmFtZTswOzA7MjQ7MTAy025vcm1hbDs
x
OzI0NjU2MTs7MTA7MTswOzkyMzQ4OTE1MjU700NhbGwgaXMgY2FuY2VsbGVkIGJ5IGluaXRpYXR
v
cjszMjA1MTUyODM50zs70zE5Mi4xNjguMi440ztzbwctNDt1bXMy0zA701BDTUESUENNVsXHNzI
5
LHR1bGVwaG9uZS1ldmVudDtxdnUwNS9yb2NpOFFLdXNlM2VpZDJ6c2I2TVlhTnliQ3B0WGRPZkd
v
dXBjTWp0a2d2dTFndDFwS3lswXZuN1NpQ1QvVhwTzVQwmtDMGFEEldTY0FjUT09CkNhbGw7MjA
x
OS4wNi4wNiAxMzoy0To0Mjs5MjM00DkxNTI10zI0NjU2MTtiaXlZay5sb2NhbC5zbWcuZ3IudHJ
1
bmsuYXV0b25hbWU7Ym15c2subG9jYwwuc3N3LmdyLnRydW5rLmF1dG9uYW1l0zA7MDs50zE2025
v
cm1hbDsxOzI0NjU2MTs7MTA7MTswOzkyMzQ4OTE1MjU700NhbGwgaXMgY2FuY2VsbGVkIGJ5IGl
u
aXRpYXRvcjszMzE0MzkyMDkw0zs70zE5Mi4xNjguMi440ztzbwctNDt1bXMy0zA701BDTUESUEN
N
VSxHNzI5LHR1bGVwaG9uZS1ldmVudDtBQTRvdUw1SUxvK1dkL0JKb0pCnNv2YVJXZFdhTWRqe1F
M
NkpyUmRiMfg2YTRxUnExSHpBQjF3UTB0a2Z3Y3p1MVRsbmZhdmdFaQkFKZ1pFanRjN0pxUT09CkN
h
bGw7MjAxOS4wNi4wNiAxMzozMDowNTs5NjUzODg5MzIyOzI0NjU3MTtiaXlZay5sb2NhbC5zbWc
u
Z3IudHJ1bmsuYXV0b25hbWU7Ym15c2subG9jYwwuc3N3LmdyLnRydW5rLmF1dG9uYW1l0zA7MDs
y
NDsxMDI7bm9ybWFsOzE7MjQ2NTcxOzcxMDsxOzA70TY1Mzg40TMymjs7Q2FsbCBpcyBjYW5jZWx
s
ZWQgYnkgaW5pdG1hdG9yOzM0MTA4MDcyMdc70zs7MTkyLjE20C4yLjg703NtZy0002VtczI7MDs
7
UENNSxQQ01VLEc3MjksdGVsZXBob25lLWV2ZW5000hYQ05PMwc5WXVqY2lJUUsrWjZPMnAySmZ
S
U2ZJT1QvSVRXN0tWUVJ0VTB2bDJnVk5UQ1p0R2k0YnB5SEsya1NFNHpPnm1NekhIY0RwMnlpUnl
H
VHVRPT0KQ2FsbDsyMDE5LjA2LjA2IDEzOjMyOjIzOzkyNTAwMTk3NDQ7MjQ1NzA402JpeXNrLmx
v
Y2FsLnNtZy5nci50cnVuay5hdXRvbmFtZTtiaXlZay5sb2NhbC5zc3cuZ3IudHJ1bmsuYXV0b25
h
bWU7MDswOzIzOzE2025vcm1hbDsxOzI0NTcwODs7MTA7MTswOzkyNTAwMTk3NDQ700NhbGwgaXM
g
Y2FuY2VsbGVkIGJ5IGluaXRpYXRvcjszOTg5MzIxMjA5MzQ5MzQ5MzQ5MzQ5MzQ5MzQ5MzQ5MzQ5
1


```

<xs:schema elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Show list of a user agents Usage: http://server:9999/commands/user_agent_list. Implementation:
      hc_user_agent_list.erl. Git: git@git.eltex.loc:httpTerminal.git. Copyright (c) 2019, Eltex. All
      rights reserved.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="ecm_common.xsd"/>
  <!-- Types -->
  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:element name="domain" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="user_agent" minOccurs="0" maxOccurs="unbounded">
              <xs:complexType>
                <xs:attribute name="name" type="xs:string" use="required"/>
                <xs:attribute name="alias" type="xs:string" use="optional"/>
                <xs:attribute name="iface" type="xs:string" use="optional"/>
                <xs:attribute name="status" type="xs:string" use="required"/>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
          <xs:attribute name="name" type="xs:string" use="required"/>
          <xs:attribute name="active_count" type="positiveIntegerType" use="required"/>
          <xs:attribute name="uniq_count" type="positiveIntegerType" use="required"/>
          <xs:attribute name="total_count" type="positiveIntegerType" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="name" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:sequence>

```

```
        </xs:complexType>
    </xs:element>
    <!-- out -->
    <xs:element name="out">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="result" type="resultType"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>
```

Пример:

Запрос: http://192.168.1.21:9999/user_agent_list

```
<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_user_agent_list.xsd">
  <request>
    <domain name="biysk.local"/>
  </request>
</in>
```

Ответ: 200


```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_user_agent_list.xsd">
  <result>
    <domain name="biysk.local" active_count="9" uniq_count="14" total_count="213">
      <user_agent name="Asterisk PBX 13.1.0~dfsg-1.1ubuntu4.1" alias="240500"
      iface="240500@biysk.local" status="false"/>
      <user_agent name="Asterisk PBX 13.1.0~dfsg-1.1ubuntu4.1" alias="240501"
      iface="240501@biysk.local" status="false"/>

      <user_agent name="Brcm-Callctrl/v1.10.3 M5T SIP Stack/4.1.2.2" alias="240462"
      iface="240462@biysk.local" status="false"/>
      <user_agent name="Brcm-Callctrl/v1.10.3 M5T SIP Stack/4.1.2.2" alias="240469"
      iface="240469@biysk.local" status="false"/>
      <user_agent name="RG-1402G/1.8.1 SN/VI0C000069 sofia-sip/1.12.10" alias="240471"
      iface="240471@biysk.local" status="false"/>
      <user_agent name="RG-1402G/1.8.1 SN/VI0C000069 sofia-sip/1.12.10" alias="240475"
      iface="240475@biysk.local" status="false"/>
      <user_agent name="RG-1404GF-W/1.11.0 SN/VI25002012 sofia-sip/1.12.10" alias="240470"
      iface="240470@biysk.local" status="false"/>
      <user_agent name="RG-1404GF-W/1.11.0 SN/VI25002012 sofia-sip/1.12.10" alias="240474"
      iface="240474@biysk.local" status="false"/>
      <user_agent name="TAU-2M.IP/2.0.0.229 SN/VI39005419 sofia-sip/1.12.10" alias="240464"
      iface="240464@biysk.local" status="false"/>
      <user_agent name="TAU-2M.IP/2.0.0.229 SN/VI39005419 sofia-sip/1.12.10" alias="240465"
      iface="240465@biysk.local" status="false"/>
      <user_agent name="Twinkle/1.9.0" alias="240007" iface="240007@biysk.local"
      status="false"/>
      <user_agent name="VP-12/1.2.1.454 SN/VI51000844 sofia-sip/1.12.10" alias="240466"
      iface="240466@biysk.local" status="false"/>
      <user_agent name="VP-12/1.2.1.454 SN/VI51000844 sofia-sip/1.12.10" alias="240473"
      iface="240473@biysk.local" status="false"/>
      <user_agent name="VP-12P/1.4.0.93 SN/VI4B000942 sofia-sip/1.12.10" alias="240467"
      iface="240467@biysk.local" status="false"/>
      <user_agent name="Yealink SIP-T28P 2.73.14.3" alias="240244"
      iface="240244@biysk.local" status="false"/>
      <user_agent name="Yealink SIP-T28P 2.73.14.3" alias="240101"
      iface="240101@biysk.local" status="false"/>
      <user_agent name="Yealink W52P 25.80.14.1" alias="240316" iface="240316@biysk.local"
      status="false"/>
    </domain>
  </result>
</out>

```

Нс_valias - Управление виртуальными алиасами в рамках виртуальной АТС

- [Hc_valias_declare](#)
- [Hc_valias_list](#)
- [Hc_valias_remove](#)

[Hc_valias_declare](#)

Команда декларирует список виртуальных алиасов в рамках определенной виртуальной АТС, с возможностью указания определенной группы интерфейсов.

Аналог команды в CoCon:

domain/<DOMAIN>/alias/virtual/declare

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/valias_declare

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />

  <!-- causeType-->
  <xs:simpleType name="causeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="already_exists"/>
      <xs:enumeration value="licence_limit"/>
      <xs:enumeration value="domain_limit"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- warningType-->
  <xs:complexType name="warningType">
    <xs:attribute name="cause" type="causeType"/>
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="interface" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- aliasType-->
  <xs:complexType name="aliasType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="interface" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="active" type="xs:boolean" use="required"/>
  </xs:complexType>

  <!-- successType-->
  <xs:complexType name="successType">
    <xs:sequence>
      <xs:element name="alias" type="aliasType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- warningsType-->
  <xs:complexType name="warningsType">
    <xs:sequence>
      <xs:element name="warning" type="warningType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- resultType-->
  <xs:complexType name="resultType">
    <xs:sequence>
      <xs:element name="success" type="successType"/>
      <xs:element name="warnings" type="warningsType"/>
    </xs:sequence>
  </xs:complexType>

```

```

    </xs:sequence>
</xs:complexType>

<!-- in-->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:attribute name="domain" type="xs:string" use="required"/>
          <xs:attribute name="context" type="xs:string" use="required"/>
          <xs:attribute name="address" type="xs:string" use="required"/>
          <xs:attribute name="group" type="xs:string" use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>

<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="result" type="resultType" />
      <xs:element name="error" type="errorType" />
    </xs:choice>
  </xs:complexType>
</xs:element>

</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/valias_declare

```

<?xml version="1.0"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_valias_declare.xsd">
  <request domain="biysk.local" context="ctx_from_local" address="240{600-602}" group="loc.gr"/>
  >
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_valias_declare.xsd">
  <result>
    <success>
      <alias domain="biysk.local" interface="virtual:240600@biysk.local" address="240600"
        active="true"/>
      <alias domain="biysk.local" interface="virtual:240601@biysk.local" address="240601"
        active="true"/>
      <alias domain="biysk.local" interface="virtual:240602@biysk.local" address="240602"
        active="true"/>
    </success>
    <warnings/>
  </result>
</out>
```

hc_valias_list

Команда возвращает список виртуальных алиасов в рамках определенного домена, для определенной группы интерфейсов.

Аналог команды в CoCon:

domain/<DOMAIN>/alias/virtual/list

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/valias_list

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа: [aliases_list.xsd](#)

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="aliases_list.xsd"/>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="domain" type="xs:string" use="required"/>
            <xs:attribute name="group" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="aliases">
          <xs:complexType>
            <xs:sequence minOccurs="0" maxOccurs="unbounded">
              <xs:element name="alias" type="aliasType" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример 1:

Запрос: http://192.168.1.21:9999/commands/valias_list

```

<?xml version="1.0" encoding="UTF-8"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_valias_list.xsd">
  <request domain="biysk.local" group="loc.gr"/>
</in>

```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_valias_list.xsd">
  <aliases>
    <alias domain="biysk.local" interface="virtual:240602@biysk.local" address="240602"
  active="true"/>
    <alias domain="biysk.local" interface="virtual:240601@biysk.local" address="240601"
  active="true"/>
    <alias domain="biysk.local" interface="virtual:240600@biysk.local" address="240600"
  active="true"/>
  </aliases>
</out>
```

Пример 2:

Запрос: http://192.168.23.34:9999/commands/valias_list

```
<?xml version="1.0" encoding="UTF-8"?>
<in
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="hc_valias_list.xsd">
  <request domain="biysk.local"/>
</in>
```

Ответ:

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="hc_valias_list.xsd">
  <aliases>
    <alias domain="biysk.local" interface="virtual:240602@biysk.local" address="240602"
  active="true"/>
    <alias domain="biysk.local" interface="virtual:240601@biysk.local" address="240601"
  active="true"/>
    <alias domain="biysk.local" interface="virtual:240600@biysk.local" address="240600"
  active="true"/>
  </aliases>
</out>
```

[Hc_valias_remove](#)

Команда удаляет список виртуальных алиасов в рамках определенного домена, с возможностью указания определенной группы интерфейсов для фильтрации.

Аналог команды в CoCon:

domain/<DOMAIN>/alias/virtual/remove

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/valias_remove

Код HTTP-ответа:

- 204 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-файла запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <!-- aliasType-->
  <xs:complexType name="aliasType">
    <xs:attribute name="domain" type="xs:string" use="required"/>
    <xs:attribute name="address" type="xs:string" use="required"/>
    <xs:attribute name="group" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- aliasesType-->
  <xs:complexType name="aliasesType">
    <xs:sequence>
      <xs:element name="alias" type="aliasType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in-->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="aliases" type="aliasesType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Пример:

Запрос:

http://192.168.1.21:9999/commands/valias_remove

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="hc_valias_remove.xsd">
  <request>
    <aliases>
      <alias domain="biysk.local" address="240601" group="loc.gr"/>
      <alias domain="biysk.local" address="240602" group="loc.gr"/>
    </aliases>
  </request>
</in>
```

Ответ: 204

Нс_yandex_tts - Управление настройками Yandex TTS

- [Нс_yandex_tts_clean](#)
- [Нс_yandex_tts_info](#)
- [Нс_yandex_tts_set](#)

[Нс_yandex_tts_clean](#)

Команда сброса настроек [Yandex TTS](#) к значениям по умолчанию

Аналог команды в CoCon:

system/tts/yandex/clean

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/yandex_tts_clean

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" />
        <xs:element name="error" type="errorType" />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/yandex_tts_clean

```

<?xml version="1.0"?>
<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="yandex_tts_clean.xsd">
  <request>
    <voice/>
  </request>
</in>

```

Ответ:

200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="yandex_tts_clean.xsd">
  <response/>
</out>

```

Ис_yandex_tts_info

Команда отображает текущие свойства Yandex TTS

Аналог команды в CoCon:

system/tts/yandex/info

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/yandex_tts_info

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />

  <xs:complexType name="constType">
    <xs:attribute name="value" type="xs:string" use="required" />
  </xs:complexType>

  <xs:complexType name="voicesType">
    <xs:sequence>
      <xs:element name="voice" type="constType" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="langsType">
    <xs:sequence>
      <xs:element name="lang" type="constType" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="emotionsType">
    <xs:sequence>
      <xs:element name="emotion" type="constType" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="infoType">
    <xs:all>
      <xs:element name="voices" type="voicesType" />
      <xs:element name="langs" type="langsType" />
      <xs:element name="emotions" type="emotionsType" />
    </xs:all>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" />
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response" type="infoType" />
        <xs:element name="error" type="errorType" />
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>
```

Пример:

Запрос:

http://192.168.1.21:9999/commands/yandex_tts_info

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="yandex_tts_info.xsd">
    <request/>
</in>
```

Ответ:

200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="yandex_tts_info.xsd">
  <response>
    <voices>
      <voice value="oksana"/>
      <voice value="jane"/>
      <voice value="omazh"/>
      <voice value="zahar"/>
      <voice value="ermil"/>
      <voice value="silaerkan"/>
      <voice value="erkanyavas"/>
      <voice value="alyss"/>
      <voice value="nick"/>
      <voice value="alena"/>
      <voice value="filipp"/>
    </voices>
    <langs>
      <lang value="ru-RU"/>
      <lang value="en-US"/>
      <lang value="tr-TR"/>
    </langs>
    <emotions>
      <emotion value="good"/>
      <emotion value="evil"/>
      <emotion value="neutral"/>
    </emotions>
  </response>
</out>

```

Ис_yandex_tts_set

Команда установки параметров Yandex TTS

Аналог команды в CoCon:

system/tts/yandex/set

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/commands/yandex_tts_set

Код HTTP-ответа:

- 200 – в случае успеха;
- 404 – в случае ошибки.

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="error.xsd" />

  <xs:complexType name="constType">
    <xs:attribute name="value" type="xs:string" use="required" />
  </xs:complexType>

  <xs:complexType name="voiceSequenceType">
    <xs:sequence>
      <xs:element name="voice" type="constType" maxOccurs="unbounded" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:all>
              <xs:element name="add" minOccurs="0" maxOccurs="1"
type="voiceSequenceType" />
              <xs:element name="remove" minOccurs="0" maxOccurs="1"
type="voiceSequenceType" />
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:all>
              <xs:element minOccurs="0" name="added" type="voiceSequenceType" />
              <xs:element minOccurs="0" name="not-added" type="voiceSequenceType" /
>
              <xs:element minOccurs="0" name="remove" type="voiceSequenceType" />
              <xs:element minOccurs="0" name="not-removed" type="voiceSequenceType"
/>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```



```
        </xs:element>
        <xs:element name="error" type="errorType" />
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
```

Пример:**Запрос:**

http://192.168.1.21:9999/commands/yandex_tts_set

```
<?xml version="1.0"?>
<in xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="yandex_tts_set.xsd">
  <request>
    <add>
      <voice value="sveta"/>
      <voice value="masha"/>
    </add>
    <remove>
      <voice value="erkanyavas"/>
    </remove>
  </request>
</in>
```

Ответ:

200

```
<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="yandex_tts_set.xsd">
  <response>
    <voices>
      <voice value="alena"/>
      <voice value="alyss"/>
      <voice value="ermil"/>
      <voice value="filipp"/>
      <voice value="jane"/>
      <voice value="masha"/>
      <voice value="nick"/>
      <voice value="oksana"/>
      <voice value="omazh"/>
      <voice value="silaerkan"/>
      <voice value="sveta"/>
      <voice value="zahar"/>
    </voices>
    <remove>
      <voice value="erkanyavas"/>
    </remove>
    <not-removed/>
    <not-added/>
    <added>
      <voice value="masha"/>
      <voice value="sveta"/>
    </added>
  </response>
</out>
```

API по работе с Call-центром

Подразделы

API call-центра поддерживает 2 типа контента:

1. XML (см. схемы приведенные в разделе "XSD файлы")
2. JSON.

Пример

Возьмем следующую структуру запроса xml:

```
POST /dom1/service/cc/arm/login HTTP/1.1

Content-Type: text/xml

host: 10.25.64.21:8086

content-length: 278

Connection: close

<?xml version='1.0' encoding='latin1'?>

<in xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"><request password_hash="av45jeAUpV+2K73T
01nCVs2bMX6E00TZI6E6DC2c0Yk=" number="2004" websocket_control="true" supervisor="false"
password_salt="ea7791ef081cbba4bd1d" login="1"/></in>
```

Она будет соответствовать следующей структуре JSON:

```
{
  "action": "login",
  "requestId": 16,
  "payload": {
    "login": "1",
    "password": "1111",
    "profile": "default",
    "number": "2004",
    "domain": "dom1",
    "role": "operator"
  }
}
```

Ответы так же будут соответствовать друг другу

```

HTTP/1.1 200 OK

connection: close

server: Cowboy

date: Tue, 15 Dec 2020 06:58:24 GMT

content-length: 336

content-type: text/xml

set-cookie: token=064803c90ec3b72b; Version=1; Path=/

<?xml version="1.0"?><out xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="login.xsd"><response version="3.14.7.391" agent_id="1" supervisor="
false" auto_logout_timeout="5"><capabilities><capability name="operator_queue_calls"/><capability
name="operator_intervention"/></capabilities></response></out>

```

```

{
  "action": "login",
  "requestId": 16,
  "payload": {
    "token":
"6601c8d19059c9b20787ec7e95f2d9d6:986d6fd21c3af83df479947f879e476ec4bb180e69c8307230828cfa5eb91b0
978d419763917579a54d5f89dcb2f0936",
    "capabilities": {
      "operatorGetInQueueCalls": true
    },
    "autoLogoutTimeout": 5,
    "agentId": "1",
    "supervisor": false
  },
  "status": 200
}

```

- Со стороны XML не поддерживаются неймспейсы и текстовый контент.

Метрики в режиме реального времени

Метрики в режиме реального времени (realtime метрики) оператора можно получить с помощью события `agent_realtime_statistics_event` (см. [agent_realtime_statistics_event.xsd](#))

Метрики с суффиксом `_t` обозначают время в секундах. Метрики с суффиксом `_c` обозначают количество в единицах.

Метрики без префиксов обозначают текущую статистику за сессию в режиме реального времени, то есть от времени входа.

- `available_t` — время, в течении которого оператор находился в состоянии "доступен";
- `auxwork_t` — время, в течении которого оператор находился "отошел";
- `blocked_t` — время, в течении которого оператор был заблокирован супервизором;
- `talk_t` — время, в течении которого оператор разговаривал;
- `acd_talk_t` — время, в течении которого оператор разговаривал, обслуживая вызовы из очередей;
- `acd_queue_talk_t` — время, в течении которого оператор разговаривал, обслуживая вызовы из очередей, включая то время, которое эти вызовы ожидали в очереди;
- `acw_t` — время, которое оператор потратил на пост-обработку вызовов;
- `accepted_c` — количество принятых разговоров, включая разговоры с очередей;
- `accepted_acd_c` — количество принятых разговоров с очередей;
- `abandoned_c` — количество пропущенных разговоров, включая разговоры с очередей;
- `abandoned_acd_c` — количество пропущенных разговоров с очередей.

Существуют также аналогичные метрики с префиксами `s_` — статистика за смену, `w_` — статистика за неделю и `m_` — статистика за месяц.

XSD файлы

Кроме HTTP ответов, указанных для той или иной команды, сервер использует следующие коды ошибок независимо от команды:

- 403 с текстовым содержимым "Permission denied" — у пользователя нет прав для выполнения данного запроса;
- 503 — команда выполняется слишком долго;
- 500 — ошибка парсинга запроса и другие внутренние ошибки сервера;
- Другие HTTP коды.

Ошибка 403 с текстом `Permission denied`, в отличие от других ошибок не должна возникать в правильно реализованном клиенте. Правильно реализованный клиент может (и должен) узнавать список возможностей и не предлагать пользователю выполнять действия, которые запрещены политикой доступа (см. [Capabilities](#) в [Команда login](#) и раздел [Специальные Разрешения](#)).

HTTP API

Подразделы

Команды HTTP API

Выполнить то или иное действие. Команды могут выполняться либо как отдельный HTTP запрос, либо по вебсокетам.

Полный список команд:

accept_fax.xsd

Команда приема факса в рамках указанного разговора.

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="AcceptFax">
    <xs:attribute name="conversation_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="accept_fax" type="AcceptFax"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Ответ:

200 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

add_participant.xsd

Команда для добавления участника в конференцию.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/add_participant

Коды ответов:

200 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="Participant">
    <xs:attribute name="number" use="optional"/> <!-- If number is specified conversation_id
will be ignored -->
    <xs:attribute name="conversation_id" use="optional"/>
  </xs:complexType>

  <xs:complexType name="AddToConference">
    <xs:all>
      <xs:element name="participants" minOccurs="0">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="participant" minOccurs="0" maxOccurs="unbounded" type="
Participant"/>
          </xs:sequence>
          <xs:attribute name="conference_id" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="add_to_conference" type="AddToConference"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>


```



```

        <xs:element name="error" type="errorType"/>
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:

Запрос:

```

0100000002750019add_participant
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "add_participant",
            "_attributes": {
              "conference_id": "1607929308885"
            },
            "_content": [
              {
                "_name": "participant",
                "_attributes": {
                  "number": "104",
                  "is_manager": "false"
                }
              }
            ]
          }
        ]
      }
    ]
  }
]

```

Код ответа:

200

Ответ:

```
0100000002240019200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "add_to_conference.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "ok"
          }
        ]
      }
    ]
  }
]
```

agent_list.xsd

Команда просмотра списка операторов группы.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/agent_list

Коды ответов:

200

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">


  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="agent.xsd"/>
  <xs:include schemaLocation="conversation.xsd"/>

  <xs:complexType name="Agents">
    <xs:sequence>
      <xs:element name="agent" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="agent_id" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="agent_id" type="xs:string"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="agent_list" type="Agents"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:

Запрос:

```
0100000000970005agent_list
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request"
      }
    ]
  }
]
```

Код ответа:

200

Ответ:

```
0100000004420005200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "agent_list.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "agent_list",
            "_content": [
              {
                "_name": "agent",
                "_attributes": {
                  "agent_id": "101"
                }
              },
              {
                "_name": "agent",
                "_attributes": {
                  "agent_id": "102"
                }
              },
              {
                "_name": "agent",
                "_attributes": {
                  "agent_id": "103"
                }
              },
              {
                "_name": "agent",
                "_attributes": {
                  "agent_id": "104"
                }
              }
            ]
          }
        ]
      }
    ]
  }
]
```

all_queues.xsd

Команда просмотра списка операторов группы.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/service/cc/arm/all_queues

Коды ответов:

200

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="Queues">
    <xs:sequence>
      <xs:element name="queue" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="queue_id" use="required"/>
          <xs:attribute name="member" use="required" type="xs:boolean"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          </xs:element>
        </xs:all>
      </xs:complexType>
    </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="queue_list" type="Queues"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответ:

200

auxwork.xsd

Команда для перехода в состояние «отошел».

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/auxwork

Коды ответов:

200 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

XSD-схема XML-запроса/ответа:


```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <!-- responseType -->
  <xs:complexType name="Auxwork">
    <xs:attribute name="reason" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="auxwork" type="Auxwork"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:
Запрос:

```
0100000001620013auxwork
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "auxwork",
            "_attributes": {
              "reason": "Busy"
            }
          }
        ]
      }
    ]
  }
]
```

Код ответа:

200

Ответ:

```
0100000002140013200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "auxwork.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "ok"
          }
        ]
      }
    ]
  }
]
```

block_operator.xsd

Команда блокировки указанного оператора.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/block_operator

Коды ответов:

200 — ошибки передаются в теле ответа.

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="BlockOperator">
    <xs:attribute name="agent_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="agent_id" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:

Запрос:

```
0100000001300013block_operator
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_attributes": {
          "agent_id": "102"
        }
      }
    ]
  }
]
```

Код ответа:

200

Ответ:

```
0100000002210013200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "block_operator.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "ok"
          }
        ]
      }
    ]
  }
]
```

call_consult.xsd

Команда подключения указанного оператора к разговору в режиме консультирования.

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <!-- agentStatusType -->
  <xs:simpleType name="Mode">
    <xs:restriction base="xs:string">
      <xs:enumeration value="consult"/>
      <xs:enumeration value="conference"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- responseType -->
  <xs:complexType name="CallConsult">
    <!--Если не указывать Id оператора, то будет использован идентификатор оператора
    запросившего конференцию-->
    <xs:attribute name="operator_id" type="xs:string" use="optional"/>
    <xs:attribute name="conversation_id" type="xs:string" use="required"/>
    <xs:attribute name="to_number" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="call_consult" type="CallConsult"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```
</xs:element>  
</xs:schema>
```

Ответы:

200

403 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

call_history.xsd

Команда просмотра истории вызовов супервизора.


```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="conversation.xsd"/>
  <xs:include schemaLocation="call_history_common.xsd"/>

  <!-- responseType -->
  <xs:complexType name="CallHistory">
    <xs:all>
      <xs:element name="url_list" type="Urllist" minOccurs="0" maxOccurs="1"/>
      <xs:element name="ss_list" type="SSList" minOccurs="0" maxOccurs="1"/>
      <xs:element name="success_faxes" type="Urllist" minOccurs="0" maxOccurs="1"/>
      <xs:element name="failed_faxes" type="Urllist" minOccurs="0" maxOccurs="1"/>
    </xs:all>
    <xs:attribute name="call_ref" type="xs:string" use="required"/>
    <xs:attribute name="digits_a" type="xs:string" use="required"/>
    <xs:attribute name="digits_b" type="xs:string" use="required"/>
    <xs:attribute name="display_name_a" type="xs:string" use="optional"/>
    <xs:attribute name="display_name_b" type="xs:string" use="optional"/>
    <xs:attribute name="agent_a_id" type="xs:string" use="optional"/>
    <xs:attribute name="agent_b_id" type="xs:string" use="optional"/>
    <xs:attribute name="start_time" type="xs:string" use="required"/> <!-- дата и время -->
    <xs:attribute name="answer_time" type="xs:string" use="optional"/> <!-- дата и время -->
    <xs:attribute name="distribution_time" type="xs:string" use="optional"/> <!-- дата и
время -->
    <xs:attribute name="release_time" type="xs:string" use="required"/> <!-- дата и время --
>
    <xs:attribute name="agent_answer_time" type="xs:string" use="optional"/> <!-- дата и
время -->
    <xs:attribute name="release_reason" type="xs:string" use="required"/>
    <xs:attribute name="duration" type="xs:nonNegativeInteger" use="required"/> <!-- время в
секундах -->
    <xs:attribute name="total_duration" type="xs:nonNegativeInteger" use="required"/> <!--
длительность в секундах -->
    <xs:attribute name="waiting_duration" type="xs:nonNegativeInteger" use="required"/> <!--
длительность в секундах -->
    <xs:attribute name="talk_duration" type="xs:nonNegativeInteger" use="required"/> <!--
длительность в секундах -->
    <xs:attribute name="queue_id" type="xs:string" use="optional"/>
    <xs:attribute name="call_record_a" type="xs:string" use="optional"/>
    <xs:attribute name="call_record_b" type="xs:string" use="optional"/>
    <xs:attribute name="original_participant_a" type="xs:string" use="optional"/>
    <xs:attribute name="original_participant_b" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:simpleType name="Mode">
    <xs:restriction base="xs:string">
      <xs:enumeration value="queues"/>
      <xs:enumeration value="agents"/>
    </xs:restriction>
  </xs:simpleType>

```

```

    </xs:restriction>
</xs:simpleType>

<!-- responseType -->
<xs:complexType name="CallHistoryList">
  <xs:sequence>
    <xs:element name="call_history" type="CallHistory" minOccurs="0" maxOccurs="unbounded
"/>
  </xs:sequence>
  <xs:attribute name="tag" type="xs:string" use="optional"/> <!-- тэг, который может быть
использован в дальнейшем для постраничного вывода -->
  <xs:attribute name="mode" type="Mode" use="optional"/> <!-- по умолчанию режим - queues -
очереди -->
</xs:complexType>

<xs:complexType name="Filters">
  <xs:attribute name="limit" type="xs:integer" use="required"/>
  <xs:attribute name="from_date" type="xs:string" use="optional"/>
  <xs:attribute name="to_date" type="xs:string" use="optional"/>
  <xs:attribute name="number" type="xs:string" use="optional"/>
  <xs:attribute name="from_number" type="xs:string" use="optional"/>
  <xs:attribute name="to_number" type="xs:string" use="optional"/>
  <xs:attribute name="cc_agent_queue" type="xs:string" use="optional"/>
  <xs:attribute name="cc_agent_group" type="xs:string" use="optional"/>
  <xs:attribute name="cc_agent_a_group" type="xs:string" use="optional"/>
  <xs:attribute name="cc_agent_b_group" type="xs:string" use="optional"/>
  <xs:attribute name="cc_agent_id" type="xs:string" use="optional"/>
  <xs:attribute name="cc_agent_a_id" type="xs:string" use="optional"/>
  <xs:attribute name="cc_agent_b_id" type="xs:string" use="optional"/>
  <xs:attribute name="cc_queue_id" type="xs:string" use="optional"/>
  <xs:attribute name="display_name" type="xs:string" use="optional"/>
  <xs:attribute name="display_name_a" type="xs:string" use="optional"/>
  <xs:attribute name="display_name_b" type="xs:string" use="optional"/>
  <xs:attribute name="skip_incomplete" type="xs:boolean" use="optional"/>
  <xs:attribute name="skip_acd" type="xs:boolean" use="optional"/> <!-- По умолчанию true
-->
  <xs:attribute name="call_ref" type="xs:string" use="optional"/>
  <xs:attribute name="tag" type="xs:string" use="optional"/> <!-- Тэг возвращенный при
предыдущем запросе. Используется для постраничного вывода -->
</xs:complexType>

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="filters" type="Filters"/>
          </xs:sequence>

```

```

        </xs:complexType>
    </xs:element>
</xs:all>
</xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
    <xs:complexType>
        <xs:choice>
            <xs:element name="response">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="history" type="CallHistoryList"/>
                    </xs:sequence>
                    <xs:attribute name="timestamp" type="xs:string"/> <!--Response timestamp
-->
                </xs:complexType>
            </xs:element>
            <xs:element name="error" type="errorType"/>
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>

```

Ответ:

200 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

call_hold.xsd

Команда для постановки разговора на удержание.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/call_hold

Коды ответов:

200

403 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <!-- responseType -->
  <xs:complexType name="CallHold">
    <!--Если не указывать Id оператора, то будет использован идентификатор оператора
    запросившего hold-->
    <xs:attribute name="operator_id" type="xs:string" use="optional"/>
    <xs:attribute name="conversation_id" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="call_hold" type="CallHold"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:

Запрос:

```
0100000001850018call_hold
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "call_hold",
            "_attributes": {
              "conversation_id": "0647e098a4b48b05"
            }
          }
        ]
      }
    ]
  }
]
```

Код ответа:

200

Ответ:

```
0100000002160018200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "call_hold.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "ok"
          }
        ]
      }
    ]
  }
]
```

call_transfer.xsd

Команда для трансфера указанного разговора.

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <!-- responseType -->
  <xs:complexType name="CallTransfer">
    <!--Если не указывать Id оператора, то будет использован идентификатор оператора
    запросившего трансфер-->
    <xs:attribute name="operator_id" type="xs:string" use="optional"/>
    <xs:attribute name="original_conversation_id" type="xs:string" use="required"/>
    <xs:attribute name="destination_conversation_id" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="call_transfer" type="CallTransfer"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Ответы:

200

403 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

call_unattended_transfer.xsd

Команда для несопровождаемого трансфера указанного разговора.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/call_unattended_transfer

Коды ответов:

200

403 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

XSD-схема XML-запроса/ответа:

```


<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <!-- responseType -->
  <xs:complexType name="CallUnattendedTransfer">
    <!--Если не указывать Id оператора, то будет использован идентификатор оператора
    запросившего трансфер-->
    <xs:attribute name="operator_id" type="xs:string" use="optional"/>
    <xs:attribute name="original_conversation_id" type="xs:string" use="required"/>
    <xs:attribute name="to_number" type="xs:string" use="required"/>
    <!--Можно опционально указывать идентификатор активного разговора вместо номера
    В этом случае трансфер будет сделан на номер участника этого разговора -->
    <xs:attribute name="destination_conversation_id" type="xs:string" use="required"/>
    <xs:attribute name="workitem_id" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="call_unattended_transfer" type="CallUnattendedTrans
fer"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```


 Пример использует технологию websocket

Пример:

Запрос:

```
0100000002180013call_unattended_transfer
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "call_unattended_transfer",
            "_attributes": {
              "conversation_id": "0647e2cf80d0cc01",
              "to_number": "104"
            }
          }
        ]
      }
    ]
  }
]
```

Код ответа:

200

Ответ:

```
0100000002310013200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "call_unattended_transfer.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "ok"
          }
        ]
      }
    ]
  }
]
]
```

call_unhold.xsd

Команда снятия разговора с удержания.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/call_unhold

Коды ответов:

200

403 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

XSD-схема XML-запроса/ответа:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <!-- responseType -->
  <xs:complexType name="CallUnhold">
    <!--Если не указывать Id оператора, то будет использован идентификатор оператора
    запросившего hold-->
    <xs:attribute name="operator_id" type="xs:string" use="optional"/>
    <xs:attribute name="conversation_id" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="call_unhold" type="CallUnhold"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

 Пример использует технологию websocket

Пример:

Запрос:

```
0100000001850019call_unhold
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "call_unhold",
            "_attributes": {
              "conversation_id": "0647e098a4b48b05"
            }
          }
        ]
      }
    ]
  }
]
```

Код ответа:

200

Ответ:

```
0100000002160019200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "call_unhold.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "ok"
          }
        ]
      }
    ]
  }
]
```

cancel_force_logout.xsd

Команда для принудительного выхода.

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="CancelForceLogout">
    <xs:attribute name="agent_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="cancel_force_logout" type="CancelForceLogout"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответ:

200

cancel_procedure.xsd

Команда отмены текущей выполняемой процедуры (используется только для отмены принудительного распределения).

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="CancelProcedure">
    <xs:attribute name="queue_id" use="required"/>
    <xs:attribute name="conversation_id" use="required"/>
    <xs:attribute name="procedure_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="cancel_procedure" type="CancelProcedure"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответы:

200

403 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

create_conference.xsd

Команда для создания конференции.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/create_conference

Коды ответов:

200

403 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

XSD-схема XML-запроса/ответа:


```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:simpleType name="Mode">
    <xs:restriction base="xs:string">
      <xs:enumeration value="public"/>
      <xs:enumeration value="private"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="Format">
    <xs:restriction base="xs:string">
      <xs:enumeration value="classic"/>
      <xs:enumeration value="consultation"/>
    </xs:restriction>
  </xs:simpleType>


  <xs:complexType name="Participant">
    <xs:attribute name="number" use="optional"/> <!-- If number is specified conversation_id
will be ignored -->
    <xs:attribute name="conversation_id" use="optional"/>
  </xs:complexType>

  <xs:complexType name="CreateConference">
    <xs:sequence>
      <xs:element name="participant" minOccurs="0" maxOccurs="unbounded" type="Participant"
/>
    </xs:sequence>
    <xs:attribute name="conference_id" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="mode" type="Mode" use="optional"/>
    <xs:attribute name="format" type="Format" use="optional"/>
    <xs:attribute name="workitem_id" use="optional"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="create_conference" type="CreateConference"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

```

```
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="ok"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="error" type="errorType"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
</xs:schema>
```

 Пример использует технологию websocket

Пример:

Запрос:

```
0100000003930016create_conference
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "create_conference",
            "_attributes": {
              "conference_id": "1607929308885",
              "description": "14:01 ",
              "mode": "public"
            },
            "_content": [
              {
                "_name": "participant",
                "_attributes": {
                  "conversation_id": "0647e2e06b635eb3"
                }
              },
              {
                "_name": "participant",
                "_attributes": {
                  "number": "104",
                  "is_manager": "false"
                }
              }
            ]
          }
        ]
      }
    ]
  }
]
}
```

Код ответа:

200

Ответ:

```
0100000002240016200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "create_conference.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "ok"
          }
        ]
      }
    ]
  }
]
```

domain_list.xsd

Команда просмотра списка доменов SSW.

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="Domain">
    <xs:attribute name="name"/>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="DomainList">
    <xs:sequence>
      <xs:element name="domain" type="Domain" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="domain_list" type="DomainList"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответ:

200

force_distribute.xsd

Команда для немедленного распределения разговора из очереди на указанного оператора.

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="ForceDistribute">
    <xs:attribute name="queue_id" use="required"/>
    <xs:attribute name="conversation_id" use="required"/>
    <xs:attribute name="operator_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="force_distribute" type="ForceDistribute"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <!-- error codes: wrong_phase, no_such_conversation, busy-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответы:

200

Ошибки, передаваемые в теле ответа:

- no_such_conversation
- wrong_phase

403

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

force_logout.xsd

Команда предназначена для выхода указанного оператора. Выход может быть отложен до завершения обработки разговоров оператора.

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="ForceLogout">
    <xs:attribute name="agent_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="force_logout" type="ForceLogout"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Ответ:

200

get_commands_list.xsd

Команда просмотра списка API команд.

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- requestType -->
  <xs:complexType name="commandType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
    <xs:attribute name="xsd" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="commands">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="command" type="commandType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Ответ:

200

group_list.xsd

Команда просмотра списков, в которых состоит текущий оператор.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/group_list

Коды ответов:

200 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

XSD-схема XML-запроса/ответа:


```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>


  <xs:complexType name="Group">
    <xs:attribute name="name"/>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="GroupList">
    <xs:sequence>
      <xs:element name="group" type="Group" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="group_list" type="GroupList"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:
Запрос:

```
010000000970005group_list
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request"
      }
    ]
  }
]
```

Код ответа:

200

Ответ:

```
0100000002890005200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "group_list.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "group_list",
            "_content": [
              {
                "_name": "group",
                "_attributes": {
                  "name": "default"
                }
              }
            ]
          }
        ]
      }
    ]
  }
]
```

integration_list.xsd

Команда позволяет получить список интеграций определенного агента КЦ.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/integration_list

Коды ответов:

200

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:complexType name="Value">
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="RedmineIntegration">
    <xs:all>
      <xs:element name="login" type="Value" minOccurs="1" maxOccurs="1"/>
      <xs:element name="password" type="Value" minOccurs="1" maxOccurs="1"/>
      <xs:element name="task_subject" type="Value" minOccurs="0" maxOccurs="1"/>
      <xs:element name="task_body" type="Value" minOccurs="0" maxOccurs="1"/>
      <xs:element name="task_project" type="Value" minOccurs="0" maxOccurs="1"/>
      <!--<xs:element name="native"/>-->
    </xs:all>
  </xs:complexType>

  <xs:complexType name="ConfluenceIntegration">
    <xs:all>
      <xs:element name="api_key" type="Value" minOccurs="1" maxOccurs="1"/>
      <!--<xs:element name="native"/>-->
    </xs:all>
  </xs:complexType>

  <xs:complexType name="CISIntegration">
    <xs:all>
      <xs:element name="request_uri" type="Value" minOccurs="1" maxOccurs="1"/>
      <!--<xs:element name="native"/>-->
    </xs:all>
  </xs:complexType>

  <xs:complexType name="SSVSIntegration">
    <xs:all>
      <xs:element name="request_uri" type="Value" minOccurs="1" maxOccurs="1"/>
      <!--<xs:element name="native"/>-->
    </xs:all>
  </xs:complexType>

  <xs:complexType name="Integrations">
    <xs:sequence>
      <xs:element name="redmine" type="RedmineIntegration" minOccurs="0" maxOccurs="1"/>
      <xs:element name="confluence" type="ConfluenceIntegration" minOccurs="0" maxOccurs="1"/>
      <xs:element name="cis" type="CISIntegration" minOccurs="0" maxOccurs="1"/>
      <xs:element name="ssvs" type="SSVSIntegration" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>

```

```

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:all>
            <xs:element name="integration_list" minOccurs="1" maxOccurs="1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="response">
        <xs:complexType>
          <xs:all>
            <xs:element name="integrations" type="Integrations" minOccurs="1" maxOccurs="1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Пример с использованием websocket:

```

curl -v 192.168.1.21:8086/biysk.local/service/cc/arm/integration_list -H 'Content-Type: text/json' --data '[{"_name":"in", "_content":[{"_name":"request", "_content":[{"_name":"integration_list"}]}]}]' --cookie token=067e132057a242b9

```

Ответ: 200

```
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "integration_list.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "integrations",
            "_content": [
              {
                "_name": "confluence",
                "_content": [
                  {
                    "_name": "access_key",
                    "_attributes": {
                      "value":
"QUJDREVGR0gXMjM0NTY3ODk6WE5DSVc5M0kyTDFUVBkU0o4MjNLMUxT0TAyS0xERk1DWlBXRU8="
                    }
                  }
                ]
              },
              {
                "_name": "redmine",
                "_content": [
                  {
                    "_name": "task_project",
                    "_attributes": {
                      "value": "SSW"
                    }
                  },
                  {
                    "_name": "login",
                    "_attributes": {
                      "value": "alex.ivanov"
                    }
                  },
                  {
                    "_name": "password",
                    "_attributes": {
                      "value": "MySuperPass"
                    }
                  }
                ]
              }
            ]
          }
        ]
      }
    ]
  }
]
```

```

    ]
  }
]
}
]
}
]
}
]

```

Пример с использованием XML:

```

curl -v http://192.168.1.21:8086/biysk.local/service/cc/arm/integration_list --data '<?xml
version="1.0" encoding="latin1"?><in xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance"><request><integration_list/></request></in>' -H 'Content-Type: text/xml; charset=UTF-8'
--cookie token=067e132057a242b9

```

Ответ: 200

```

<?xml version="1.0"?>
<out
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
  xs:noNamespaceSchemaLocation="integration_list.xsd">
  <response>
    <integrations>
      <confluence>
        <access_key value="QUJDREVGROgxMjM0NTY3ODk6WE5DSVc5M0kyTDFUVBkU0o4MjNLMUxTOTAyS0
xERK1DWlBXRU8="/>
      </confluence>
      <redmine>
        <task_project value="SSW"/>
        <login value="alex.ivanov"/>
        <password value="MySuperPass"/>
      </redmine>
    </integrations>
  </response>
</out>

```

journal_list.xsd

Команда просмотра списка событий Call-центра.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:complexType name="JournalList">
    <xs:attribute name="subsystem" type="xs:string" use="required"/>
    <xs:attribute name="event" type="xs:string" use="required"/> <!-- имя события или * -->
    <xs:attribute name="entity" type="xs:string" use="required"/> <!-- имя сущности или * --
  >

    <xs:attribute name="before_timestamp" type="xs:string" use="required"/> <!--YY/MM/DD
HH:MM:SS-->
    <xs:attribute name="limit" type="xs:positiveInteger" use="required"/> <!-- max 50 -->
  </xs:complexType>

  <xs:complexType name="Event">
    <xs:sequence>
      <xs:element name="properties" minOccurs="0" maxOccurs="1">
        <xs:complexType>
          <xs:anyAttribute/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="subsystem" type="xs:string" use="required"/>
    <xs:attribute name="event" type="xs:string" use="required"/>
    <xs:attribute name="entity" type="xs:string" use="required"/>
    <xs:attribute name="timestamp" type="xs:string" use="required"/> <!--YY/MM/DD HH:MM:SS-->
  </xs:complexType>

  <xs:complexType name="Events">
    <xs:sequence>
      <xs:element name="event" type="Event" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="journal_list" maxOccurs="unbounded" type="JournalLi
st"/>
            </xs:sequence>
            <xs:attribute name="domain" type="xs:string"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```



```

</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="events" minOccurs="0" maxOccurs="unbounded" type="E
vents"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Ответ:

200

leave_conference.xsd

Команды выхода из конференции. Для владельца конференции это значит — перестать получать нотификации о конференции.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/leave_conference

Коды ответов:

200

403 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <!-- responseType -->
  <xs:complexType name="LeaveConference">
    <xs:attribute name="conference_id" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="leave_conference" type="LeaveConference"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:
Запрос:

```
0100000001870021leave_conference
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "leave_conference",
            "_attributes": {
              "conference_id": "1607929308885"
            }
          }
        ]
      }
    ]
  }
]
```

Код ответа:

200

Ответ:

```
0100000002230021200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "leave_conference.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "ok"
          }
        ]
      }
    ]
  }
]
```

make_agent_call.xsd

Команда выполнения вызова по ID оператора.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/make_agent_call

Коды ответов:

200 — в случае успеха;

403 — оператор, который пытается выполнить команду, не зарегистрирован на телефонном аппарате.

409 — эта команда уже выполняется для данного оператора.

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">


  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="MakeAgentCall">
    <xs:attribute name="agent_id" use="required"/>
    <xs:attribute name="workitem_id" use="optional"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="make_agent_call" type="MakeAgentCall"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok">
                <xs:complexType>
                  <xs:attribute name="internal_call_id" type="xs:string"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:

Запрос:

```
0100000001710012make_agent_call
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "make_agent_call",
            "_attributes": {
              "agent_id": "104"
            }
          }
        ]
      }
    ]
  }
]
```

Код ответа:

200

Ответ:

```

0100000002760012200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "make_agent_call.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "ok",
            "_attributes": {
              "internal_call_id": "0647e1420a26ad29"
            }
          }
        ]
      }
    ]
  }
]

```

make_available.xsd

Команда перехода в состояние «доступен».

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/make_available

Коды ответов:

200

403 — оператор, который пытается выполнить команду, не зарегистрирован на телефонном аппарате;

404 — такого оператора нет или он не зарегистрирован;

409 — эта команда уже выполняется для данного оператора.

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="make_available"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:
Запрос:


```
0100000001370014make_available
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "make_available"
          }
        ]
      }
    ]
  }
]
```

Код ответа:

403

Ответ:

```
{
  "action": "logout",
  "requestId": 0,
  "payload": {},
  "status": 403
}
```

monitor.xsd

Команда для подписки на событие.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/monitor

Коды ответов:

200

404 — Подписка на несуществующее событие

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- baseEventType -->
  <xs:complexType name="baseEventType" abstract="true">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="entity" type="xs:string" use="optional"/>
    <xs:attribute name="timestamp" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- monitorEventType -->
  <xs:complexType name="monitorEventType">
    <xs:complexContent>
      <xs:extension base="baseEventType"/>
    </xs:complexContent>
  </xs:complexType>
  <!-- startType -->
  <xs:complexType name="startType">
    <xs:sequence>
      <xs:element name="event" type="monitorEventType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- stopType -->
  <xs:complexType name="stopType">
    <xs:sequence>
      <xs:element name="event" type="monitorEventType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- monitor -->
  <xs:element name="monitor">
    <xs:complexType>
      <xs:all>
        <xs:element name="start" type="startType" minOccurs="0"/>
        <xs:element name="stop" type="stopType" minOccurs="0"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:

Запрос:

```
0100000004250011monitor
[
  {
    "_name": "monitor",
    "_content": [
      {
        "_name": "start",
        "_content": [
          {
            "_name": "event",
            "_attributes": {
              "name": "conversations_event",
              "entity": "101"
            }
          }
        ],
      },
      {
        "_name": "event",
        "_attributes": {
          "name": "conversations_event",
          "entity": "102"
        }
      },
      {
        "_name": "event",
        "_attributes": {
          "name": "conversations_event",
          "entity": "103"
        }
      },
      {
        "_name": "event",
        "_attributes": {
          "name": "conversations_event",
          "entity": "104"
        }
      }
    ]
  }
]
]
```

Код ответа:

200

Ответ:

```
0100000000190011200
```

`move_conversation_to_queue.xsd`

Команда для перемещения разговора из одной очереди в другую.

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="MoveConversationToQueue">
    <xs:attribute name="conversation_id" use="required"/>
    <xs:attribute name="destination_queue_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="move_conversation_to_queue" type="MoveConversationT
oQueue"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <!-- error codes: wrong_phase, no_such_conversation-->
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответы:

200

403 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

operator_call_history.xsd

Команда просмотра истории вызовов текущего оператора.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/operator_call_history

Коды ответов:

200

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>
  <xs:include schemaLocation="conversation.xsd"/>
  <xs:include schemaLocation="call_history_common.xsd"/>

  <xs:complexType name="CallHistory">
    <xs:all>
      <xs:element name="url_list" type="Urllist" minOccurs="0" maxOccurs="1"/>
      <xs:element name="ss_list" type="SSList" minOccurs="0" maxOccurs="1"/>
      <xs:element name="success_faxes" type="Urllist" minOccurs="0" maxOccurs="1"/>
      <xs:element name="failed_faxes" type="Urllist" minOccurs="0" maxOccurs="1"/>
    </xs:all>
    <xs:attribute name="call_ref" type="xs:string" use="required"/>
    <xs:attribute name="start_time" type="xs:string" use="required"/>
    <xs:attribute name="answer_time" type="xs:string" use="optional"/>
    <xs:attribute name="release_time" type="xs:string" use="required"/>
    <xs:attribute name="duration" type="xs:nonNegativeInteger" use="required"/> <!--
длительность в секундах -->
    <xs:attribute name="total_duration" type="xs:nonNegativeInteger" use="required"/> <!--
длительность в секундах -->
    <xs:attribute name="waiting_duration" type="xs:nonNegativeInteger" use="required"/> <!--
длительность в секундах -->
    <xs:attribute name="talk_duration" type="xs:nonNegativeInteger" use="required"/> <!--
длительность в секундах -->
    <xs:attribute name="queue_id" type="xs:string" use="optional"/> <!-- в текущей версии
если вызов не проходил через очередь
                                                    то queue_id = "", но
в новой версии мы от этого избавимся -->
    <xs:attribute name="direction" type="Direction" use="required"/>
    <xs:attribute name="display_name" type="xs:string" use="required"/> <!-- отображаемое имя
оператора, для данного вызова в истории -->
    <xs:attribute name="digits" type="xs:string" use="required"/> <!-- номер оператора, для
данного вызова в истории -->
    <xs:attribute name="call_record" type="xs:string" use="optional"/> <!-- запись разговора
со стороны оператора -->
    <xs:attribute name="remote_display_name" type="xs:string" use="required"/>
    <xs:attribute name="remote_digits" type="xs:string" use="optional"/>
    <xs:attribute name="remote_agent_id" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- responseType -->
  <xs:complexType name="CallHistoryList">
    <xs:sequence>
      <xs:element name="call_history" type="CallHistory" minOccurs="0" maxOccurs="unbounded
"/>
    </xs:sequence>
    <xs:attribute name="tag" type="xs:string" use="optional"/> <!-- тэг, который может быть
использован в дальнейшем для страничного вывода -->

```



```

</xs:complexType>

<xs:complexType name="Filters">
  <xs:attribute name="offset" type="xs:integer" use="required"/>
  <xs:attribute name="limit" type="xs:integer" use="required"/>
  <xs:attribute name="from_date" type="xs:string" use="optional"/>
  <xs:attribute name="to_date" type="xs:string" use="optional"/>
  <xs:attribute name="number" type="xs:string" use="optional"/>
  <xs:attribute name="from_number" type="xs:string" use="optional"/>
  <xs:attribute name="to_number" type="xs:string" use="optional"/>
  <xs:attribute name="cc_agent_group" type="xs:string" use="optional"/>
  <xs:attribute name="cc_agent_a_group" type="xs:string" use="optional"/>
  <xs:attribute name="cc_agent_b_group" type="xs:string" use="optional"/>
  <xs:attribute name="cc_agent_id" type="xs:string" use="optional"/>
  <xs:attribute name="cc_agent_a_id" type="xs:string" use="optional"/>
  <xs:attribute name="cc_agent_b_id" type="xs:string" use="optional"/>
  <xs:attribute name="display_name" type="xs:string" use="optional"/>
  <xs:attribute name="display_name_a" type="xs:string" use="optional"/>
  <xs:attribute name="display_name_b" type="xs:string" use="optional"/>
  <xs:attribute name="skip_incomplete" type="xs:boolean" use="optional"/>
  <xs:attribute name="direction" type="Direction" use="optional"/> <!-- Направление вызова.
Входящий/исходящий -->
  <xs:attribute name="tag" type="xs:string" use="optional"/> <!-- Тэг возвращенный при
предыдущем запросе. Используется для постраничного вывода -->
</xs:complexType>

<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:all>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="filters" type="Filters"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:choice>
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="history" type="CallHistoryList"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:choice>
  </xs:complexType>
</xs:element>

```

```

-->
        <xs:attribute name="timestamp" type="xs:string"/> <!--Response timestamp
-->
        </xs:complexType>
    </xs:element>
    <xs:element name="error" type="errorType"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:

Запрос:

```

0100000001590012operator_call_history
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "filters",
            "_attributes": {
              "limit": "50"
            }
          }
        ]
      }
    ]
  }
]

```

Код ответа:

200

Ответ:

```

0100000058100012200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "operator_call_history.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_attributes": {
          "timestamp": "2020/12/10 14:25:53"
        },
        "_content": [
          {
            "_name": "history",
            "_attributes": {
              "tag":
                "g2gEZAADdGFnZAAHaG1zdG9yewGcZAAIZGF0ZXRpbwVoAmgDYgAAB+RhDGEKaANhDmEHYThsAAAABmgCZAALY2NfYWd1bnRf
                aWRtAAAAAzEwMmgCZAAPc2tpcF9pbmNvbXBsZXRlZAAEdHJlZWwCZAAQc2tpcF9zdXB1cnZpc2luZ2QABHRYdWVoAmQACmNhb
                GxzX3R5cGVkAAadoaXN0b3J5aAJKAAR2aWV3aAJKAAhvcGVyYXRvcu0AAAADMTAyaAJkAAZhY3RpdmVkaAVmYwzZW0="
            },
            "_content": [
              {
                "_name": "call_history",
                "_attributes": {
                  "total_duration": "4",
                  "waiting_duration": "4",
                  "talk_duration": "0",
                  "release_reason": "released_by_system",
                  "digits": "102",
                  "remote_digits": "101",
                  "call_ref": "613295627",
                  "duration": "0",
                  "release_time": "2020/12/10 14:20:54",
                  "answer_time": "",
                  "start_time": "2020/12/10 14:20:50",
                  "direction": "in",
                  "call_record": "",
                  "remote_display_name": "..... 1",
                  "display_name": "",
                  "remote_agent_id": ""
                },
                "_content": [
                  {
                    "_name": "url_list"
                  }
                ]
              }
            ]
          }
        ]
      }
    ]
  }
]

```

```

},
{
  "_name": "call_history",
  "_attributes": {
    "total_duration": "13",
    "waiting_duration": "7",
    "talk_duration": "6",
    "release_reason": "released_by_system",
    "digits": "102",
    "remote_digits": "101",
    "call_ref": "537644738",
    "duration": "5",
    "release_time": "2020/12/10 14:20:45",
    "answer_time": "2020/12/10 14:20:39",
    "start_time": "2020/12/10 14:20:32",
    "direction": "in",
    "call_record": "",
    "remote_display_name": "..... 1",
    "display_name": "",
    "remote_agent_id": ""
  },
  "_content": [
    {
      "_name": "url_list"
    }
  ]
},
{
  "_name": "call_history",
  "_attributes": {
    "total_duration": "7",
    "waiting_duration": "4",
    "talk_duration": "3",
    "release_reason": "released_by_b",
    "digits": "102",
    "remote_digits": "101",
    "call_ref": "3834028254196687921",
    "duration": "3",
    "release_time": "2020/12/10 14:20:26",
    "answer_time": "2020/12/10 14:20:23",
    "start_time": "2020/12/10 14:20:19",
    "direction": "out",
    "call_record": "",
    "remote_display_name": "..... 1",
    "display_name": "",
    "remote_agent_id": ""
  },
  "_content": [
    {
      "_name": "url_list"
    }
  ]
}

```

```

    }
  ],
},
{
  "_name": "call_history",
  "_attributes": {
    "total_duration": "3",
    "waiting_duration": "3",
    "talk_duration": "0",
    "release_reason": "released_by_system",
    "digits": "102",
    "remote_digits": "101",
    "call_ref": "4115547623",
    "duration": "0",
    "release_time": "2020/12/10 14:17:44",
    "answer_time": "",
    "start_time": "2020/12/10 14:17:41",
    "direction": "in",
    "call_record": "",
    "remote_display_name": "..... 1",
    "display_name": "",
    "remote_agent_id": ""
  },
  "_content": [
    {
      "_name": "url_list"
    }
  ]
},
{
  "_name": "call_history",
  "_attributes": {
    "total_duration": "10",
    "waiting_duration": "5",
    "talk_duration": "5",
    "release_reason": "released_by_b",
    "digits": "102",
    "remote_digits": "101",
    "call_ref": "4060321042",
    "duration": "5",
    "release_time": "2020/12/10 14:17:38",
    "answer_time": "2020/12/10 14:17:33",
    "start_time": "2020/12/10 14:17:28",
    "direction": "in",
    "call_record": "",
    "remote_display_name": "..... 1",
    "display_name": "",
    "remote_agent_id": ""
  },
  "_content": [

```

```

    {
      "_name": "url_list"
    }
  ]
},
{
  "_name": "call_history",
  "_attributes": {
    "total_duration": "9",
    "waiting_duration": "5",
    "talk_duration": "4",
    "release_reason": "released_by_a",
    "digits": "102",
    "remote_digits": "101",
    "call_ref": "7162187274952783718",
    "duration": "4",
    "release_time": "2020/12/10 14:17:22",
    "answer_time": "2020/12/10 14:17:18",
    "start_time": "2020/12/10 14:17:13",
    "direction": "out",
    "call_record": "",
    "remote_display_name": "..... 1",
    "display_name": "",
    "remote_agent_id": ""
  },
  "_content": [
    {
      "_name": "url_list"
    }
  ]
},
{
  "_name": "call_history",
  "_attributes": {
    "total_duration": "3",
    "waiting_duration": "3",
    "talk_duration": "0",
    "release_reason": "released_by_system",
    "digits": "102",
    "remote_digits": "101",
    "call_ref": "3255665922",
    "duration": "0",
    "release_time": "2020/12/10 14:14:19",
    "answer_time": "",
    "start_time": "2020/12/10 14:14:16",
    "direction": "in",
    "call_record": "",
    "remote_display_name": "..... 1",
    "display_name": "",
    "remote_agent_id": ""
  }
}

```

```

},
  "_content": [
    {
      "_name": "url_list"
    }
  ]
},
{
  "_name": "call_history",
  "_attributes": {
    "total_duration": "11",
    "waiting_duration": "8",
    "talk_duration": "3",
    "release_reason": "released_by_b",
    "digits": "102",
    "remote_digits": "101",
    "call_ref": "3171756862",
    "duration": "2",
    "release_time": "2020/12/10 14:14:07",
    "answer_time": "2020/12/10 14:14:04",
    "start_time": "2020/12/10 14:13:56",
    "direction": "in",
    "call_record": "",
    "remote_display_name": "..... 1",
    "display_name": "",
    "remote_agent_id": ""
  },
  "_content": [
    {
      "_name": "url_list"
    }
  ]
},
{
  "_name": "call_history",
  "_attributes": {
    "total_duration": "12",
    "waiting_duration": "5",
    "talk_duration": "7",
    "release_reason": "released_by_a",
    "digits": "102",
    "remote_digits": "101",
    "call_ref": "7291666691949541221",
    "duration": "6",
    "release_time": "2020/12/10 14:13:43",
    "answer_time": "2020/12/10 14:13:36",
    "start_time": "2020/12/10 14:13:31",
    "direction": "out",
    "call_record": "",
    "remote_display_name": "..... 1",

```

```

        "display_name": "",
        "remote_agent_id": ""
    },
    "_content": [
        {
            "_name": "url_list"
        }
    ]
},
{
    "_name": "call_history",
    "_attributes": {
        "total_duration": "8",
        "waiting_duration": "4",
        "talk_duration": "4",
        "release_reason": "released_by_a",
        "digits": "102",
        "remote_digits": "101",
        "call_ref": "3991933534187108144",
        "duration": "3",
        "release_time": "2020/12/10 14:08:45",
        "answer_time": "2020/12/10 14:08:41",
        "start_time": "2020/12/10 14:08:37",
        "direction": "out",
        "call_record": "",
        "remote_display_name": "..... 1",
        "display_name": "",
        "remote_agent_id": ""
    },
    "_content": [
        {
            "_name": "url_list"
        }
    ]
},
{
    "_name": "call_history",
    "_attributes": {
        "total_duration": "19",
        "waiting_duration": "3",
        "talk_duration": "16",
        "release_reason": "released_by_system",
        "digits": "102",
        "remote_digits": "101",
        "call_ref": "1661436472",
        "duration": "15",
        "release_time": "2020/12/10 14:08:15",
        "answer_time": "2020/12/10 14:07:59",
        "start_time": "2020/12/10 14:07:56",
        "direction": "in",

```



```
        "call_record": "",
        "remote_display_name": "..... 1",
        "display_name": "",
        "remote_agent_id": ""
    },
    "_content": [
        {
            "_name": "url_list"
        }
    ]
}
]
```

ping.xsd

Команда проверки активности текущей сессии. Продлить время жизни текущей сессии.

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Ответ:

200 — pong

queue_list.xsd

Команда просмотра списка очередей.

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="Queues">
    <xs:sequence>
      <xs:element name="queue" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="queue_id" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="group_id" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="queue_list" type="Queues"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответ:

200

read_user_data.xsd

Команда просмотра кастомных свойств пользователя.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/read_user_data

Коды ответов:

200 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="ReadUserData">
    <xs:attribute name="agent_id" use="required"/>
    <xs:attribute name="application_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="read_user_data" type="ReadUserData"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="user_data">
                <xs:complexType>
                  <xs:sequence>
                    <xs:any namespace="##any" processContents="lax" minOccurs="
0" maxOccurs="unbounded" />
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:

Запрос:

```
0100000001780049read_user_data
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "read_user_data",
            "_attributes": {
              "application_id": "hodor"
            }
          }
        ]
      }
    ]
  }
]
```

Код ответа:

200

Ответ:

```

0100000007290049200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "read_user_data.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "user_data",
            "_content": [
              {
                "_name": "user_data",
                "_attributes": {
                  "user_data": "{\myReports\": [{\"id\": \"1607932267415\", \"name\": \"new_test\",
\\plotDef\": [{\"id\": 58019194, \"chartType\": \"line\", \"visible\": true, \"name\":
\\\".....\", \"queueIds\": [], \"queueMetricIds\": [], \"operatorIds\":
[\"102\"], \"operatorMetricIds\": [\"incomingCallsCount\"]}], \"dateFrom\":
\\\"2020-12-07T00:00:00.000+07:00\", \"dateTo\": \"2020-12-14T23:59:59.000+07:00\", \"dynamicDate\":
\\\"week\", \"intervalType\": \"auto\"}]}"
                }
              ]
            ]
          }
        ]
      }
    ]
  }
]

```

reject_consultation.xsd

Команда отклонения консультирования указанного разговора.

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="RejectConsultation">
    <xs:attribute name="conversation_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="reject_consultation" type="RejectConsultation"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

reject_fax.xsd

Команда отклонения приема факса.

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="RejectFax">
    <xs:attribute name="conversation_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="reject_fax" type="RejectFax"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответ:

200 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

reject_supervising.xsd

Команда отклонения супервизинга указанного разговора.

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="RejectSupervising">
    <xs:attribute name="conversation_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="reject_supervising" type="RejectSupervising"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

remove_conference.xsd

Команда удаления указанной конференции.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/remove_conference

Коды ответов:

200

403 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <!-- responseType -->
  <xs:complexType name="RemoveConference">
    <!--Если не указывать Id оператора, то будет использован идентификатор оператора
    запросившего конференцию-->
    <xs:attribute name="operator_id" type="xs:string" use="optional"/>
    <xs:attribute name="conference_id" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="remove_conference" type="RemoveConference"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:

Запрос:

```
0100000001880024remove_conference
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "remove_conference",
            "_attributes": {
              "conference_id": "1607931249619"
            }
          }
        ]
      }
    ]
  }
]
```

Код ответа:

200

Ответ:

```
0100000002240024200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "remove_conference.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "ok"
          }
        ]
      }
    ]
  }
]
```

`remove_conversation_from_queue.xsd`

Команда удаления разговора из очереди.

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="RemoveConversationFromQueue">
    <xs:attribute name="conversation_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="remove_conversation_from_queue" type="RemoveConvers
ationFromQueue"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:choice>
              <xs:element name="ok"/>
              <xs:element name="error">
                <xs:complexType>
                  <xs:attribute name="reason">
                    <xs:simpleType>
                      <xs:restriction base="xs:string">
                        <xs:enumeration value="no_such_conversation"/>
                      </xs:restriction>
                    </xs:simpleType>
                  </xs:attribute>
                </xs:complexType>
              </xs:element>
            </xs:choice>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```
</xs:schema>
```

Ответы:

200

403 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

remove_participant.xsd

Команда удаления участника конференции.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/remove_participant

Коды ответов:

200

403 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

XSD-схема XML-запроса/ответа:


```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="RemoveParticipant">
    <xs:attribute name="conference_id" type="xs:string" use="required"/>
    <xs:attribute name="participant_id" type="xs:string" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="remove_participant" type="RemoveParticipant"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:
Запрос:

```
0100000002100017remove_participant
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "remove_participant",
            "_attributes": {
              "conference_id": "1607929308885",
              "participant_id": "0"
            }
          }
        ]
      }
    ]
  }
]
```

Код ответа:

200

Ответ:

```
0100000002250017200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "remove_participant.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "ok"
          }
        ]
      }
    ]
  }
]
```

`remove_user_data.xsd`

Команда удаления кастомных данных пользователя.

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="RemoveUserData">
    <xs:attribute name="agent_id" use="required"/>
    <xs:attribute name="application_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="remove_user_data" type="RemoveUserData"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответ:

200 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

save_user_data.xsd

Команда сохранения кастомных данных пользователя.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/save_user_data

Коды ответов:

200 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- cannot_perform_cmd

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="SaveUserData">
    <xs:sequence>
      <xs:element name="user_data" minOccurs="0" maxOccurs="1">
        <xs:complexType>
          <xs:sequence>
            <xs:any namespace="##any" processContents="lax" minOccurs="0" maxOccurs="
unbounded" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="agent_id" use="optional"/>
    <xs:attribute name="application_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="save_user_data" type="SaveUserData"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

⚠ Пример использует технологию websocket

Пример:

Запрос:

```
0100000007180050save_user_data
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "save_user_data",
            "_attributes": {
              "application_id": "hodor"
            },
            "_content": [
              {
                "_name": "user_data",
                "_content": [
                  {
                    "_name": "user_data",
                    "_attributes": {
                      "user_data": "{\"myReports\": [{\"id\": \"1607932267415\", \"name\": \"new_test_new\", \"plotDef\": [{\"id\": 58019194, \"chartType\": \"line\", \"visible\": true, \"name\": \".....\", \"queueIds\": [], \"queueMetricIds\": [], \"operatorIds\": [\"102\"], \"operatorMetricIds\": [\"incomingCallsCount\"]}], \"dateFrom\": \"2020-12-07T00:00:00.000+07:00\", \"dateTo\": \"2020-12-14T23:59:59.000+07:00\", \"dynamicDate\": \"week\", \"intervalType\": \"auto\"}]}"
                    }
                  ]
                }
            ]
          }
        ]
      }
    ]
  }
]

```

Код ответа:

200

Ответ:

```
0100000002210050200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "save_user_data.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "ok"
          }
        ]
      }
    ]
  }
]
```

send_to_email.xsd

Команда для отправки принятого факса на e-mail.


```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="SendToEmailType">
    <xs:attribute name="subject" use="required" type="xs:string"/>
    <xs:attribute name="body" use="required" type="xs:string"/>
    <xs:attribute name="to" use="required" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The list of the email addresses, separated by comma.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="links" use="required" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">
          The list of the attachment's links, separated by comma.
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="send_to_email" type="SendToEmailType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

```

```
        </xs:choice>
    </xs:complexType>
</xs:element>
</xs:schema>
```

Ответы:

200

404 — Ошибки передаются в теле ответа

Ошибки, передаваемые в теле ответа:

- send_email_error
- file_not_found
- cannot_perform_cmd

statistics_show.xsd

Команда просмотра статистики.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/statistics_show

Коды ответов:

200

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- systemType -->
  <xs:simpleType name="systemType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="ecss10.common"/>
      <xs:enumeration value="ecss10.ssw"/>
      <xs:enumeration value="ecss10.cc"/>
      <xs:enumeration value="ecss10.tc"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- statisticTypeType -->
  <xs:simpleType name="statisticTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="counter"/>
      <xs:enumeration value="gauge"/>
      <xs:enumeration value="meter"/>
      <xs:enumeration value="histogram"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- statisticsValueType -->
  <xs:complexType name="statisticsValueType">
    <xs:attribute name="time" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="value" type="xs:int"/>
    <xs:attribute name="n" type="xs:int"/>
    <xs:attribute name="min" type="xs:int" use="optional"/>
    <xs:attribute name="max" type="xs:int" use="optional"/>
    <xs:attribute name="avg" type="xs:double" use="optional"/>
  </xs:complexType>
  <!-- statisticsInType -->
  <xs:complexType name="statisticsInType">
    <xs:attribute name="from" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="to" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="system" type="systemType"/>
    <xs:attribute name="subsystem" type="xs:string" use="optional"/>
    <xs:attribute name="statistic" type="xs:string" use="optional"/>
    <xs:attribute name="type" type="statisticTypeType" use="optional"/>
    <xs:attribute name="variable_part_1" type="xs:string" use="optional"/>
    <xs:attribute name="variable_part_2" type="xs:string" use="optional"/>
    <xs:attribute name="measure_type" type="statisticIntervalType" use="optional"/>
  </xs:complexType>
  <!-- statisticsOutType -->
  <xs:complexType name="statisticsOutType">
    <xs:sequence>
      <xs:element name="value" minOccurs="0" maxOccurs="unbounded" type="statisticsValueTyp
e"/>
    </xs:sequence>
    <xs:attribute name="system" type="systemType"/>

```

```

<xs:attribute name="subsystem" type="xs:string"/>
<xs:attribute name="statistic" type="xs:string"/>
<xs:attribute name="type" type="statisticTypeType"/>
<xs:attribute name="measure_type" type="xs:string" use="optional"/>
<xs:attribute name="sum" type="xs:float" use="optional"/>
<xs:attribute name="variable_part_1" type="xs:string" use="optional"/>
<xs:attribute name="variable_part_2" type="xs:string" use="optional"/>
</xs:complexType>
<!-- statisticIntervalType -->
<xs:simpleType name="statisticIntervalType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="realtime"/>
    <xs:enumeration value="intrahour"/>
    <xs:enumeration value="hour"/>
    <xs:enumeration value="day"/>
    <xs:enumeration value="week"/>
    <xs:enumeration value="month"/>
    <xs:enumeration value="auto"/>
  </xs:restriction>
</xs:simpleType>
<!-- in -->
<xs:element name="in">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="request">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="statistics" maxOccurs="unbounded" type="statisticsI
nType"/>
          </xs:sequence>
          <xs:attribute name="domain" type="xs:string"/>
          <xs:attribute name="measure_type" type="statisticIntervalType"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- out -->
<xs:element name="out">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="response">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="statistics" minOccurs="0" maxOccurs="unbounded"
type="statisticsOutType"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```
    </xs:complexType>  
  </xs:element>  
</xs:schema>
```

XSD-схема с общими типами данных **statistics_common.xsd**

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <!-- systemType -->
  <xs:simpleType name="systemType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="ecss10.common"/>
      <xs:enumeration value="ecss10.ssw"/>
      <xs:enumeration value="ecss10.cc"/>
      <xs:enumeration value="ecss10.tc"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- statisticTypeType -->
  <xs:simpleType name="statisticTypeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="counter"/>
      <xs:enumeration value="gauge"/>
      <xs:enumeration value="meter"/>
      <xs:enumeration value="histogram"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- statisticIntervalType -->
  <xs:simpleType name="statisticIntervalType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="realtime"/>
      <xs:enumeration value="intrahour"/>
      <xs:enumeration value="hour"/>
      <xs:enumeration value="day"/>
      <xs:enumeration value="week"/>
      <xs:enumeration value="month"/>
      <xs:enumeration value="auto"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- dynamicDateType -->
  <xs:simpleType name="dynamicDateType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="none"/>
      <xs:enumeration value="today"/>
      <xs:enumeration value="yesterday"/>
      <xs:enumeration value="week"/>
      <xs:enumeration value="month"/>
      <xs:enumeration value="quarter"/>
      <xs:enumeration value="year"/>
    </xs:restriction>
  </xs:simpleType>
</xs:schema>
```

```

<!-- statisticsInType -->
<xs:complexType name="statisticsInType">
  <xs:attribute name="system" type="systemType" use="required"/>
  <xs:attribute name="subsystem" type="xs:string"/>
  <xs:attribute name="statistic" type="xs:string"/>
  <xs:attribute name="type" type="statisticTypeType"/>
  <xs:attribute name="variable_part_1" type="xs:string"/>
  <xs:attribute name="variable_part_2" type="xs:string"/>
  <!-- Displayable metric name -->
  <xs:attribute name="title" type="xs:string"/>
</xs:complexType>

<!-- statisticRequestType -->
<xs:complexType name="statisticRequestType">
  <xs:sequence>
    <xs:element name="statistics" maxOccurs="unbounded" type="statisticsInType"/>
  </xs:sequence>
  <xs:attribute name="from" type="xs:dateTime" use="required"/>
  <xs:attribute name="to" type="xs:dateTime" use="required"/>
  <xs:attribute name="measure_type" type="statisticIntervalType"/>
  <xs:attribute name="request_time_utc" type="xs:boolean"/>
</xs:complexType>

<!-- statisticReportType -->
<xs:complexType name="statisticReportType">
  <xs:sequence>
    <xs:element name="chart" maxOccurs="unbounded">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="statistics" maxOccurs="unbounded"
type="statisticsInType"/>
        </xs:sequence>
        <xs:attribute name="name" type="xs:string" use="required"/>
        <xs:attribute name="chart_type" type="xs:string"/>
        <xs:attribute name="visible" type="xs:boolean"/>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="dynamic_date" type="dynamicDateType" use="required"/>
  <!-- Required if dynamic_date == 'none' -->
  <xs:attribute name="from" type="xs:dateTime"/>
  <xs:attribute name="to" type="xs:dateTime"/>
  <xs:attribute name="id" type="xs:string"/>
  <xs:attribute name="measure_type" type="statisticIntervalType"/>
  <xs:attribute name="description" type="xs:string"/>
  <xs:attribute name="created_at" type="xs:dateTime"/>
  <xs:attribute name="task_id" type="xs:string"/>
</xs:complexType>

```

```
</xs:schema>
```

 Пример использует технологию websocket

Пример:

Запрос:

```
0100000003440027statistics_show
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_content": [
          {
            "_name": "statistics",
            "_attributes": {
              "from": "2020/12/07 00:00:00",
              "to": "2020/12/14 23:59:59",
              "statistic": "accepted_direct_in_c",
              "subsystem": "agent",
              "system": "ecss10.cc",
              "type": "counter",
              "measure_type": "auto",
              "variable_part_2": "102"
            }
          }
        ]
      }
    ]
  }
]
```

Код ответа:

200

Ответ:


```

0100000013900027200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "statistics_show.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "statistics",
            "_attributes": {
              "domain": "arko",
              "from": "2020/12/07 00:00:00",
              "to": "2020/12/14 23:59:59",
              "system": "ecss10.cc",
              "subsystem": "agent",
              "statistic": "accepted_direct_in_c",
              "type": "counter",
              "measure_type": "day",
              "sum": "6.000000",
              "variable_part_2": "102"
            },
            "_content": [
              {
                "_name": "value",
                "_attributes": {
                  "time": "2020/12/07 23:59:59",
                  "value": "0",
                  "n": "-",
                  "min": "-",
                  "max": "-",
                  "avg": "-"
                }
              },
              {
                "_name": "value",
                "_attributes": {
                  "time": "2020/12/08 23:59:59",
                  "value": "0",
                  "n": "-",
                  "min": "-",
                  "max": "-",
                  "avg": "-"
                }
              }
            ],
          }
        ],
      }
    ],
  },
]

```

```
{
  "_name": "value",
  "_attributes": {
    "time": "2020/12/09 23:59:59",
    "value": "0",
    "n": "-",
    "min": "-",
    "max": "-",
    "avg": "-"
  }
},
{
  "_name": "value",
  "_attributes": {
    "time": "2020/12/10 23:59:59",
    "value": "5",
    "n": "-",
    "min": "-",
    "max": "-",
    "avg": "-"
  }
},
{
  "_name": "value",
  "_attributes": {
    "time": "2020/12/11 23:59:59",
    "value": "1",
    "n": "-",
    "min": "-",
    "max": "-",
    "avg": "-"
  }
},
{
  "_name": "value",
  "_attributes": {
    "time": "2020/12/12 23:59:59",
    "value": "0",
    "n": "-",
    "min": "-",
    "max": "-",
    "avg": "-"
  }
},
{
  "_name": "value",
  "_attributes": {
    "time": "2020/12/13 23:59:59",
    "value": "0",
    "n": "-",
```

```
        "min": "-",
        "max": "-",
        "avg": "-"
    }
},
{
    "_name": "value",
    "_attributes": {
        "time": "2020/12/14 23:59:59",
        "value": "-",
        "n": "-",
        "min": "-",
        "max": "-",
        "avg": "-"
    }
}
]
}
]
}
]
}
```

supervise.xsd

Команда подключения к разговору в режиме супервизинга.

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:simpleType name="SupervisingType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="conference"/>
      <xs:enumeration value="consult"/>
      <xs:enumeration value="observing"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- responseType -->
  <xs:complexType name="Supervise">
    <xs:attribute name="conversation_id" type="xs:string" use="required"/>
    <xs:attribute name="conversation_ref" type="xs:string" use="required"/>
    <xs:attribute name="mode" type="SupervisingType" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="supervise" type="Supervise"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответ:

200

unlock_operator.xsd

Команда разблокировки указанного оператора.

Метод HTTP-запроса:

POST

Шаблон HTTP-запроса:

http://host:port/<DOMAIN>/service/cc/arm/unlock_operator

Коды ответов:

200

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:attribute name="agent_id" use="required"/>
            </xs:complexType>
          </xs:element>
        </xs:all>
      </xs:complexType>
    </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

 Пример использует технологию websocket

Пример:
Запрос:

```
0100000001300014unlock_operator
[
  {
    "_name": "in",
    "_content": [
      {
        "_name": "request",
        "_attributes": {
          "agent_id": "102"
        }
      }
    ]
  }
]
```

Код ответа:

200

Ответ:

```
0100000002230014200
[
  {
    "_name": "out",
    "_attributes": {
      "xmlns:xs": "http://www.w3.org/2001/XMLSchema-instance",
      "xs:noNamespaceSchemaLocation": "unlock_operator.xsd"
    },
    "_content": [
      {
        "_name": "response",
        "_content": [
          {
            "_name": "ok"
          }
        ]
      }
    ]
  }
]
```

События HTTP API

После подписки на то или иное событие с помощью команды `monitor`, по вебсокетам начинают приходить нотификации.

Полный список событий:

`agent_info_event.xsd`

Событие, содержащее информацию об операторе.


```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>
  <xs:include schemaLocation="agent.xsd"/>

  <xs:complexType name="AgentInfoEventType">
    <xs:sequence>
      <xs:element name="agent_info" type="AgentInfo"/>
    </xs:sequence>
  </xs:complexType>

  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:sequence>
            <xs:element name="agent_info" type="AgentInfoEventType"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

</xs:schema>
```

agent_list_event.xsd

Событие, содержащее информацию об операторах группы.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>

  <xs:simpleType name="StatusType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="stopped"/>
      <xs:enumeration value="available"/>
      <xs:enumeration value="auxwork"/>
      <xs:enumeration value="removed"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="LineStatusType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="idle"/>
      <xs:enumeration value="alerting"/>
      <xs:enumeration value="talking"/>
      <xs:enumeration value="acw"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="Agent">
    <xs:attribute name="agent_id" type="xs:string"/>
    <xs:attribute name="status" type="StatusType"/>
    <xs:attribute name="line_status" type="LineStatusType"/>
    <xs:attribute name="is_supervisor" type="xs:boolean"/>
    <xs:attribute name="display_name" type="xs:string" use="optional"/>
    <xs:attribute name="number" type="xs:string" use="optional"/>
    <xs:attribute name="auxwork_reason" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="GroupAgentList">
    <xs:sequence>
      <xs:element name="agent" type="Agent"/>
    </xs:sequence>
    <xs:attribute name="group_id" type="xs:string"/>
  </xs:complexType>

  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:sequence>
            <xs:element name="group" type="GroupAgentList"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

```

```
        </xs:complexContent>  
    </xs:complexType>  
</xs:element>  
  
</xs:schema>
```

agent_realtime_statistics_event2.xsd

Ещё один вариант события, содержащего статистику оператора в режиме реального времени. В отличие от первого содержит ещё и информацию о метриках, которые в данный момент растут.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>

  <xs:complexType name="Metric">
    <xs:attribute name="value" type="xs:string"/>
    <xs:attribute name="increment" type="xs:boolean"/> <!--Растёт ли в данный момент эта
метрика-->
  </xs:complexType>

  <xs:complexType name="Metrics">
    <xs:all>
      <xs:element name="available_t" type="Metric"/>
      <xs:element name="auxwork_t" type="Metric"/>
      <xs:element name="blocked_t" type="Metric"/>
      <xs:element name="talk_t" type="Metric"/>
      <xs:element name="acd_talk_t" type="Metric"/>
      <xs:element name="acd_queue_talk_t" type="Metric"/>
      <xs:element name="acw_t" type="Metric"/>
      <xs:element name="accepted_c" type="Metric"/>
      <xs:element name="accepted_acd_c" type="Metric"/>
      <xs:element name="abandoned_c" type="Metric"/>
      <xs:element name="abandoned_acd_c" type="Metric"/>
    </xs:all>
  </xs:complexType>

  <xs:complexType name="AgentRealtimeStatisticsEventType">
    <xs:sequence>
      <xs:element name="session" type="Metrics"/>
      <xs:element name="shift" type="Metrics"/>
      <xs:element name="week" type="Metrics"/>
      <xs:element name="month" type="Metrics"/>
    </xs:sequence>
    <xs:attribute name="agent_id" type="xs:string"/>
  </xs:complexType>

  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:sequence>
            <xs:element name="agent_realtime_statistics" type="AgentRealtimeStatistic
sEventType"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>

```

```
</xs:element>  
  
</xs:schema>
```

agent_realtime_statistics_event.xsd

Событие, содержащее статистику оператора в режиме реального времени.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>

  <xs:simpleType name="LimitType">
    <xs:union>
      <xs:simpleType>
        <xs:restriction base="xs:nonNegativeInteger"/>
      </xs:simpleType>
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="unbounded"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:union>
  </xs:simpleType>

  <xs:complexType name="AgentRealtimeStatisticsEventType">
    <xs:attribute name="agent_id" type="xs:string"/>
    <xs:attribute name="available_t" type="xs:nonNegativeInteger"/> <!-- время, которое
оператор находился в состоянии "доступен" -->
    <xs:attribute name="auxwork_t" type="xs:nonNegativeInteger"/> <!-- время, которое
оператор находился "отошел" -->
    <xs:attribute name="blocked_t" type="xs:nonNegativeInteger"/> <!-- время, которое
оператор был заблокирован супервизором -->
    <xs:attribute name="talk_t" type="xs:nonNegativeInteger"/> <!-- время, которое оператор
разговаривал -->
    <xs:attribute name="acd_talk_t" type="xs:nonNegativeInteger"/> <!-- время, которое
оператор разговаривал, обслуживая вызовы из очередей -->
    <xs:attribute name="acd_queue_talk_t" type="xs:nonNegativeInteger"/> <!-- время, которое
оператор разговаривал, обслуживая вызовы из очередей, включая то время, которое эти вызовы
ждали в очереди -->
    <xs:attribute name="acw_t" type="xs:nonNegativeInteger"/> <!-- время, которое оператор
потратил на пост-обработку вызовов -->
    <xs:attribute name="accepted_c" type="xs:nonNegativeInteger"/> <!-- количество принятых
разговоров, включая разговоры с очередями -->
    <xs:attribute name="accepted_acd_c" type="xs:nonNegativeInteger"/> <!-- количество
принятых разговоров с очередями -->
    <xs:attribute name="abandoned_c" type="xs:nonNegativeInteger"/> <!-- количество
пропущенных разговоров, включая разговоры с очередями -->
    <xs:attribute name="abandoned_acd_c" type="xs:nonNegativeInteger"/> <!-- количество
пропущенных разговоров с очередями -->
    <!-- метрики за смену -->
    <xs:attribute name="s_available_t" type="xs:nonNegativeInteger"/>
    <xs:attribute name="s_auxwork_t" type="xs:nonNegativeInteger"/>
    <xs:attribute name="s_blocked_t" type="xs:nonNegativeInteger"/>
    <xs:attribute name="s_talk_t" type="xs:nonNegativeInteger"/>

```

```

<xs:attribute name="s_acd_talk_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="s_acd_queue_talk_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="s_acw_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="s_accepted_c" type="xs:nonNegativeInteger"/>
<xs:attribute name="s_accepted_acd_c" type="xs:nonNegativeInteger"/>
<xs:attribute name="s_abandoned_c" type="xs:nonNegativeInteger"/>
<xs:attribute name="s_abandoned_acd_c" type="xs:nonNegativeInteger"/>
<!-- метрики за неделю -->
<xs:attribute name="w_available_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="w_auxwork_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="w_blocked_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="w_talk_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="w_acd_talk_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="w_acd_queue_talk_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="w_acw_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="w_accepted_c" type="xs:nonNegativeInteger"/>
<xs:attribute name="w_accepted_acd_c" type="xs:nonNegativeInteger"/>
<xs:attribute name="w_abandoned_c" type="xs:nonNegativeInteger"/>
<xs:attribute name="w_abandoned_acd_c" type="xs:nonNegativeInteger"/>
<!-- метрики за месяц -->
<xs:attribute name="m_available_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="m_auxwork_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="m_blocked_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="m_talk_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="m_acd_talk_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="m_acd_queue_talk_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="m_acw_t" type="xs:nonNegativeInteger"/>
<xs:attribute name="m_accepted_c" type="xs:nonNegativeInteger"/>
<xs:attribute name="m_accepted_acd_c" type="xs:nonNegativeInteger"/>
<xs:attribute name="m_abandoned_c" type="xs:nonNegativeInteger"/>
<xs:attribute name="m_abandoned_acd_c" type="xs:nonNegativeInteger"/>
</xs:complexType>

<xs:element name="event">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="baseEventType">
        <xs:sequence>
          <xs:element name="agent_realtime_statistics" type="AgentRealtimeStatistic
sEventType"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:element>

</xs:schema>

```

auxwork_profile_event.xsd

Схема предназначена:

- для агента, чтобы он мог выбирать причину технического перерыва с последующим переходом в технический перерыв. Свой профиль агент должен знать по подписке на событие agent_info_event.
 - для супервизора, чтобы он мог переводить произвольного агента в технический перерыв с заданной причиной. Профиль агента супервизор может знать по подписке на конфигурацию очередей.
-


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="monitor.xsd"/>
  <!-- event -->
  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:choice>
            <xs:element name="full" type="fullType"/>
            <xs:element name="partial" type="partialType"/>
          </xs:choice>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

  <!-- resultFullType -->
  <xs:complexType name="fullType">
    <xs:sequence>
      <xs:element name="profile" type="profileType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- resultPartialType -->
  <xs:complexType name="partialType">
    <xs:sequence>
      <xs:element name="action" type="actionType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- actionType -->
  <xs:complexType name="actionType">
    <xs:sequence>
      <xs:element name="profile" type="profileType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="type" type="actionTypeContent"/>
  </xs:complexType>

  <!-- profileType -->
  <xs:complexType name="profileType">
    <xs:sequence maxOccurs="unbounded">
      <xs:element name="reason" type="reasonType"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string"/>
  </xs:complexType>

  <!-- reasonType -->

```

```
<xs:complexType name="reasonType">
  <xs:attribute name="name" type="xs:string"/>
  <xs:attribute name="en" type="xs:string"/>
  <xs:attribute name="ru" type="xs:string"/>
</xs:complexType>

<!-- actionTypeContent -->
<xs:simpleType name="actionTypeContent">
  <xs:restriction base="xs:string">
    <xs:enumeration value="added"/>
    <xs:enumeration value="removed"/>
    <xs:enumeration value="changed"/>
  </xs:restriction>
</xs:simpleType>
</xs:schema>
```

Примеры:**full:**

```

<?xml version="1.0"?>
<event name="queue_profile_event" timestamp="1970/01/01 07:00:00">
  <full>
    <profile name="test">
      <reason name="124" en="124" ru="124"/>
      <reason name="busy" en="Busy" ru="Занят"/>
      <reason name="callback" en="Callback" ru="Обратный звонок"/>
      <reason name="dinner" en="Dinner" ru="Обед"/>
      <reason name="handle_call" en="Call handling" ru="Обработка вызова"/>
      <reason name="locality_transfer" en="Transferring to another locality" ru="Передача в
другой населенный пункт"/>
      <reason name="not_available" en="" ru=""/>
      <reason name="rest" en="Rest" ru="Отдых"/>
    </profile>
    <profile name="new">
      <reason name="busy" en="Busy" ru="Занят"/>
      <reason name="custom_reason" en="" ru=""/>
    </profile>
    <profile name="default">
      <reason name="busy" en="Busy" ru="Занят"/>
    </profile>
    <profile name="103">
      <reason name="busy" en="Busy" ru="Занят"/>
      <reason name="dinner" en="Dinner" ru="Обед"/>
      <reason name="124" en="124" ru="124"/>
      <reason name="handle_call" en="Call handling" ru="Обработка вызова"/>
      <reason name="callback" en="Callback" ru="Обратный звонок"/>
      <reason name="rest" en="Rest" ru="Отдых"/>
      <reason name="locality_transfer" en="Transferring to another locality" ru="Передача в
другой населенный пункт"/>
    </profile>
  </full>
</event>

```

partial:

```
<?xml version="1.0"?>
<event name="queue_profile_event" timestamp="1970/01/01 07:00:00">
  <partial>
    <action type="changed">
      <profile name="profile2">
        <reason name="busy" en="Busy" ru="Занят"/>
        <reason name="callback" en="Callback" ru="Обратный звонок"/>
      </profile>
    </action>
    <action type="removed">
      <profile name="profile3">
        <reason name="busy" en="Busy" ru="Занят"/>
      </profile>
    </action>
    <action type="added">
      <profile name="profile1">
        <reason name="busy" en="Busy" ru="Занят"/>
      </profile>
    </action>
  </partial>
</event>
```

conference_participants_event2.xsd

Вторая версия нотификации об участниках конференции. В отличие от первой версии здесь гарантируется, что у участника, возвращенного в конференцию повторно, будет сохраняться тот же самый идентификатор.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>

  <xs:simpleType name="StatusType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="connecting"/>
      <xs:enumeration value="connected"/>
      <xs:enumeration value="disconnected"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="ReasonType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="failed"/>
      <xs:enumeration value="released"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="ParticipantInfo">
    <xs:attribute name="id" type="xs:string"/>
    <xs:attribute name="status" type="StatusType"/>
    <xs:attribute name="digits" type="xs:string"/>
    <xs:attribute name="agent_id" type="xs:string" use="optional"/>
    <xs:attribute name="display_name" type="xs:string" use="optional"/>
    <xs:attribute name="disconnect_reason" type="ReasonType" use="optional"/> <!-- only if
status == disconnected -->
    <xs:anyAttribute/>
  </xs:complexType>

  <xs:complexType name="Participants">
    <xs:sequence>
      <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="participant" type="ParticipantInfo"/>
      </xs:choice>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string"/>
  </xs:complexType>

  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:sequence>
            <xs:element name="participants" type="Participants"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

```

```
        </xs:complexContent>  
    </xs:complexType>  
</xs:element>  
  
</xs:schema>
```

conference_participants_event.xsd

Нотификация о статусе участников конференции.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>

  <xs:complexType name="ParticipantNew">
    <xs:attribute name="id" type="xs:string"/>
  </xs:complexType>

  <xs:simpleType name="StatusType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="connecting"/>
      <xs:enumeration value="connected"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="ReasonType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="failed"/>
      <xs:enumeration value="released"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="ParticipantInfo">
    <xs:attribute name="id" type="xs:string"/>
    <xs:attribute name="status" type="StatusType"/>
    <xs:anyAttribute/>
  </xs:complexType>

  <xs:complexType name="ParticipantRemoved">
    <xs:attribute name="id" type="xs:string"/>
    <xs:attribute name="reason" type="ReasonType"/>
    <xs:anyAttribute/>
  </xs:complexType>

  <xs:complexType name="Participants">
    <xs:sequence>
      <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="new" type="ParticipantNew"/>
        <xs:element name="info" type="ParticipantInfo"/>
        <xs:element name="removed" type="ParticipantRemoved"/>
      </xs:choice>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string"/>
  </xs:complexType>

  <xs:element name="event">
    <xs:complexType>

```

```
<xs:complexContent>
  <xs:extension base="baseEventType">
    <xs:sequence>
      <xs:element name="participants" type="Participants"/>
    </xs:sequence>
  </xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>

</xs:schema>
```

conferences_event.xsd

Нотификация с информацией о конференции.


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>

  <xs:complexType name="ConferenceNew">
    <xs:attribute name="id" type="xs:string"/>
    <xs:attribute name="is_owner" type="xs:boolean"/>
    <xs:attribute name="start_time" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="ConferenceInfo">
    <xs:attribute name="id" type="xs:string"/>
    <xs:anyAttribute/>
  </xs:complexType>

  <xs:complexType name="ConferenceRemoved">
    <xs:attribute name="id" type="xs:string"/>
  </xs:complexType>

  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:choice>
            <xs:element name="new" type="ConferenceNew"/>
            <xs:element name="info" type="ConferenceInfo"/>
            <xs:element name="removed" type="ConferenceRemoved"/>
          </xs:choice>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

conversation.xsd

Нотификация с информацией о разговорах оператора.

XSD-схема XML-нотификации:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <!--AutoCompleteTimeout-->
  <xs:simpleType name="AutoCompleteTimeout">
    <xs:restriction base="xs:nonNegativeInteger">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="3600"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="ConversationStatus">
    <xs:restriction base="xs:string">
      <xs:enumeration value="alerting"/>
      <xs:enumeration value="initiating"/>
      <xs:enumeration value="working"/>
      <xs:enumeration value="fax_receiving"/>
      <xs:enumeration value="fax_rejecting"/>
      <xs:enumeration value="hold"/>
      <xs:enumeration value="acw"/>
      <xs:enumeration value="released"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="ConversationType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="call"/>
      <xs:enumeration value="mail"/>
      <xs:enumeration value="chat"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="Direction">
    <xs:restriction base="xs:string">
      <xs:enumeration value="in"/>
      <xs:enumeration value="out"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="Conversation">
    <xs:sequence>
      <xs:element name="additional_info" minOccurs="0" maxOccurs="1">
        <xs:complexType>
          <xs:sequence>
            <xs:any namespace="##any" processContents="lax" minOccurs="0" maxOccurs="
unbounded" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>


```

```

        </xs:element>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="type" type="ConversationType" use="required"/>
    <xs:attribute name="status" type="ConversationStatus" use="required"/>
    <xs:attribute name="start_time" type="xs:string" use="required"/>
    <xs:attribute name="answer_time" type="xs:string" use="optional"/>
    <xs:attribute name="release_time" type="xs:string" use="optional"/>
    <xs:attribute name="fax_receiving_time" type="xs:string" use="optional"/>
    <xs:attribute name="acd" type="xs:boolean" use="required"/>
    <xs:attribute name="queue_id" type="xs:string" use="optional"/>
    <xs:attribute name="queue_description" type="xs:string" use="optional"/>
    <xs:attribute name="direction" type="Direction" use="required"/>
    <xs:attribute name="display_name_a" type="xs:string" use="required"/>
    <xs:attribute name="display_name_b" type="xs:string" use="required"/>
    <xs:attribute name="digits_a" type="xs:string" use="optional"/>
    <xs:attribute name="digits_b" type="xs:string" use="optional"/>
    <xs:attribute name="fax_receive" type="xs:boolean" use="optional"/>
    <xs:attribute name="ref" type="xs:string" use="required"/>
    <xs:attribute name="auto_complete_launched" type="xs:boolean" use="required"/>
    <xs:attribute name="auto_complete_timeout" type="AutoCompleteTimeout" use="optional"/>
    <xs:attribute name="description" type="xs:string" use="optional"/>
    <xs:attribute name="is_consult" type="xs:boolean" use="optional"/>
    <xs:attribute name="is_conference" type="xs:boolean" use="optional"/>
    <xs:attribute name="on_consultation" type="xs:boolean" use="optional"/>
    <xs:attribute name="intervention" type="xs:boolean" use="optional"/>
    <xs:attribute name="workitem_id" type="xs:string" use="optional"/>
</xs:complexType>

</xs:schema>

```

 Пример использует технологию websocket

Пример:

 В примере так же используется схема [conversations_event.xsd](#)

Уведомление по подписке:

```

0100000006639999200
[
  {
    "_name": "event",
    "_attributes": {
      "name": "conversations_event",
      "timestamp": "2020/12/15 09:32:03",
      "entity": "104"
    },
    "_content": [
      {
        "_name": "conversations",
        "_content": [
          {
            "_name": "conversation",
            "_attributes": {
              "id": "0647f406344ff68b",
              "call_ref": "3689684372030578994",
              "type": "call",
              "status": "alerting",
              "acd": "true",
              "direction": "in",
              "ref": "{<<\0647f4061e402b4d\>>,182}",
              "is_consult": "false",
              "display_name_a": "..... 1",
              "display_name_b": "",
              "fax_receive": "true",
              "digits_b": "104",
              "digits_a": "101",
              "queue_id": "test_line",
              "queue_description": "",
              "start_time": "2020/12/15 09:32:03",
              "intervention": "true"
            },
            "_content": [
              {
                "_name": "additional_info"
              }
            ]
          }
        ]
      }
    ]
  }
]

```

conversations_event.xsd

Нотификация с информацией о разговорах оператора.

XSD-схема XML-нотификации:


```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>
  <xs:include schemaLocation="conversation.xsd"/>


  <xs:complexType name="ConversationsEventType">
    <xs:sequence>
      <xs:element name="conversation" type="Conversation" minOccurs="0" maxOccurs="unbounde
d"/>
    </xs:sequence>
  </xs:complexType>

  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:sequence>
            <xs:element name="conversations" type="ConversationsEventType"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

</xs:schema>
```

 Пример использует технологию websocket

Пример:

 В примере так же используется схема [conversation.xsd](#)

Уведомление по подписке:

```

0100000006639999200
[
  {
    "_name": "event",
    "_attributes": {
      "name": "conversations_event",
      "timestamp": "2020/12/15 09:32:03",
      "entity": "104"
    },
    "_content": [
      {
        "_name": "conversations",
        "_content": [
          {
            "_name": "conversation",
            "_attributes": {
              "id": "0647f406344ff68b",
              "call_ref": "3689684372030578994",
              "type": "call",
              "status": "alerting",
              "acd": "true",
              "direction": "in",
              "ref": "{<<\0647f4061e402b4d\>>,182}",
              "is_consult": "false",
              "display_name_a": "..... 1",
              "display_name_b": "",
              "fax_receive": "true",
              "digits_b": "104",
              "digits_a": "101",
              "queue_id": "test_line",
              "queue_description": "",
              "start_time": "2020/12/15 09:32:03",
              "intervention": "true"
            },
            "_content": [
              {
                "_name": "additional_info"
              }
            ]
          }
        ]
      }
    ]
  }
]

```

group_list_event.xsd

Подписка на список групп Call-центра


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="monitor.xsd"/>
  <!-- event -->
  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:choice>
            <xs:element name="full" type="fullType"/>
            <xs:element name="partial" type="partialType"/>
          </xs:choice>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

  <!-- resultFullType -->
  <xs:complexType name="fullType">
    <xs:sequence>
      <xs:element name="group" type="groupType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- resultPartialType -->
  <xs:complexType name="partialType">
    <xs:sequence>
      <xs:element name="action" type="actionType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- actionType -->
  <xs:complexType name="actionType">
    <xs:sequence>
      <xs:element name="group" type="groupType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="type" type="actionTypeContent"/>
  </xs:complexType>

  <!-- groupType -->
  <xs:complexType name="groupType">
    <xs:sequence maxOccurs="unbounded">
      <xs:choice>
        <xs:element name="agent" type="agentType"/>
      </xs:choice>
    </xs:sequence>
    <xs:attribute name="group_id" type="xs:string"/>
  </xs:complexType>

```

```
<!-- agentType -->
<xs:complexType name="agentType">
  <xs:attribute name="agent_id" type="xs:string"/>
  <xs:attribute name="is_supervisor" type="xs:boolean"/>
</xs:complexType>

<!-- actionTypeContent -->
<xs:simpleType name="actionTypeContent">
  <xs:restriction base="xs:string">
    <xs:enumeration value="added"/>
    <xs:enumeration value="removed"/>
    <xs:enumeration value="changed"/>
  </xs:restriction>
</xs:simpleType>
</xs:schema>
```

group_realtime_statistics_event.xsd

Схема подписки на изменения статистики группы:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>

  <xs:complexType name="GroupRealtimeStatisticsEventType">
    <xs:sequence>
      <xs:element name="queue" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="id" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="group_id" type="xs:string" use="required"/>
    <xs:attribute name="start" type="xs:dateTime" use="required"/>
    <xs:attribute name="finish" type="xs:dateTime" use="required"/>
    <!-- Значения на момент обращения (нестатистические данные) -->
    <xs:attribute name="total_agent_c" type="xs:string"/> <!-- количество операторов в группе
-->
    <xs:attribute name="available_agent_c" type="xs:string"/> <!-- количество залогиненых
операторов в группе -->
    <xs:attribute name="auxwork_agent_c" type="xs:string"/> <!-- количество операторов в
группе не на рабочем месте -->
    <xs:attribute name="blocked_agent_c" type="xs:string"/> <!-- количество заблокированных
супервизором операторов в группе -->
    <xs:attribute name="talking_agent_c" type="xs:string"/> <!-- количество операторов в
группе, обрабатывающих вызовы -->
    <xs:attribute name="acw_agent_c" type="xs:string"/> <!-- количество операторов в группе,
занимающихся пост-обработкой вызова -->
    <!-- Статистика за день -->
    <xs:attribute name="max_talk_t" type="xs:string"/> <!-- максимальное время обработки
вызовов, в секундах -->
    <xs:attribute name="avg_talk_t" type="xs:string"/> <!-- среднее время обработки вызовов,
в секундах -->
    <xs:attribute name="incoming_c" type="xs:string"/> <!-- количество входящих вызовов,
включая разговоры с очередей -->
    <xs:attribute name="direct_in_c" type="xs:string"/> <!-- количество входящих вызовов, не
включая разговоры с очередей -->
  </xs:complexType>

  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:sequence>
            <xs:element name="group_realtime_statistics" type="GroupRealtimeStatistic
sEventType"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

```

```
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
</xs:element>

</xs:schema>
```

queue_config_event.xsd

Схема подписки на состояние очередей для супервизора.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:include schemaLocation="monitor.xsd"/>
  <!-- event -->
  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:choice>
            <xs:element name="full" type="fullType"/>
            <xs:element name="partial" type="partialType"/>
          </xs:choice>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

  <!-- resultFullType -->
  <xs:complexType name="fullType">
    <xs:sequence>
      <xs:element name="profile" type="profileType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- resultPartialType -->
  <xs:complexType name="partialType">
    <xs:sequence>
      <xs:element name="action" type="actionType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- actionType -->
  <xs:complexType name="actionType">
    <xs:sequence>
      <xs:element name="profile" type="profileType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="type" type="actionTypeContent"/>
  </xs:complexType>

  <!-- profileType -->
  <xs:complexType name="profileType">
    <xs:sequence maxOccurs="unbounded">
      <xs:element name="reason" type="reasonType"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string"/>
  </xs:complexType>

  <!-- reasonType -->

```

```

<xs:complexType name="reasonType">
  <xs:attribute name="name" type="xs:string"/>
  <xs:attribute name="en" type="xs:string"/>
  <xs:attribute name="ru" type="xs:string"/>
</xs:complexType>

<!-- actionTypeContent -->
<xs:simpleType name="actionTypeContent">
  <xs:restriction base="xs:string">
    <xs:enumeration value="added"/>
    <xs:enumeration value="removed"/>
    <xs:enumeration value="changed"/>
  </xs:restriction>
</xs:simpleType>
</xs:schema>

```

Примеры:**full:**

```

<?xml version="1.0"?>
<event name="queue_config_event" timestamp="1970/01/01 07:00:00">
  <full>
    <queue queue_id="queue2" description="">
      <operators>
        <operator operator_id="101" type="alias"/>
      </operators>
      <groups>
        <group group_id="group2"/>
        <group group_id="group1"/>
      </groups>
    </queue>
    <queue queue_id="queue1" description="some description">
      <operators>
        <operator operator_id="564" type="agent"/>
      </operators>
      <groups>
        <group group_id="default"/>
      </groups>
    </queue>
  </full>
</event>

```

partial:

```

<?xml version="1.0"?>
<event name="queue_config_event" timestamp="1970/01/01 07:00:00">
  <partial>
    <action type="added">
      <queue queue_id="queue3" description="">
        <operators/>
        <groups>
          <group group_id="group1"/>
        </groups>
      </queue>
    </action>
    <action type="removed">
      <queue queue_id="queue4" description="">
        <operators/>
        <groups>
          <group group_id="group2"/>
        </groups>
      </queue>
    </action>
    <action type="changed">
      <queue queue_id="queue5" description="changed description">
        <operators>
          <operator operator_id="564" type="agent"/>
        </operators>
        <groups>
          <group group_id="changed_group"/>
        </groups>
      </queue>
    </action>
  </partial>
</event>

```

queue_conversations_event2.xsd

Вторая версия события со списком разговоров в очереди. Присылаются только измененные разговоры. При изначальной подписке присылается полный список разговоров. Далее только изменения.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>

  <xs:simpleType name="Status">
    <xs:restriction base="xs:string">
      <xs:enumeration value="waiting"/>
      <xs:enumeration value="distributing"/>
      <xs:enumeration value="distributed"/>
      <xs:enumeration value="released"/>
      <xs:enumeration value="removed"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="QueueConversation">
    <xs:attribute name="conversation_id" type="xs:integer"/>
    <xs:attribute name="position" type="xs:integer"/>
    <xs:attribute name="priority" type="xs:integer"/>
    <xs:attribute name="status" type="Status"/>
    <xs:attribute name="seize_time" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="distribution_time" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="answer_time" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="release_time" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="digits_a" type="xs:string"/>
    <xs:attribute name="display_name_a" type="xs:string"/>
    <xs:attribute name="digits_b" type="xs:string"/> <!-- Телефонный номер агента -->
    <xs:attribute name="display_name_b" type="xs:string"/> <!-- Отображаемое имя агента -->
    <xs:attribute name="workitem_id" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="QueueConversatinsEventType">
    <xs:sequence>
      <xs:element name="conversation" type="QueueConversation" minOccurs="0" maxOccurs="unb
ounded"/>
    </xs:sequence>
    <xs:attribute name="queue_id" type="xs:string"/>
  </xs:complexType>

  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:sequence>
            <xs:element name="queue_conversations2" type="QueueConversatinsEventType"
/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

```



```
        </xs:complexContent>  
    </xs:complexType>  
</xs:element>  
  
</xs:schema>
```

queue_conversations_event.xsd

Событие со списком разговоров в очереди. Раз в 3 секунды приходит полный список разговоров очереди.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>

  <xs:simpleType name="Status">
    <xs:restriction base="xs:string">
      <xs:enumeration value="waiting"/>
      <xs:enumeration value="distributing"/>
      <xs:enumeration value="distributed"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="QueueConversation">
    <xs:attribute name="conversation_id" type="xs:integer"/>
    <xs:attribute name="position" type="xs:integer"/>
    <xs:attribute name="priority" type="xs:integer"/>
    <xs:attribute name="status" type="Status"/>
    <xs:attribute name="seize_time" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="distribution_time" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="answer_time" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="release_time" type="xs:string"/> <!--YY/MM/DD HH:MM:SS-->
    <xs:attribute name="digits_a" type="xs:string"/>
    <xs:attribute name="display_name_a" type="xs:string"/>
    <xs:attribute name="digits_b" type="xs:string"/> <!-- Телефонный номер агента -->
    <xs:attribute name="display_name_b" type="xs:string"/> <!-- Отображаемое имя агента -->
    <xs:attribute name="workitem_id" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="QueueConversatinsEventType">
    <xs:sequence>
      <xs:element name="conversation" type="QueueConversation" minOccurs="0" maxOccurs="unb
ounded"/>
    </xs:sequence>
    <xs:attribute name="queue_id" type="xs:string"/>
  </xs:complexType>

  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:sequence>
            <xs:element name="queue_conversations" type="QueueConversatinsEvent"/
>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>

```

```
</xs:element>  
  
</xs:schema>
```

queue_list_event.xsd

Оператор видит только свои очереди. Подписка на список очередей для оператора.
Оператор с правами супервизора видит все очереди.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>

  <!-- event -->
  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:choice>
            <xs:element name="full" type="fullType"/>
            <xs:element name="partial" type="partialType"/>
          </xs:choice>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

  <!-- resultFullType -->
  <xs:complexType name="fullType">
    <xs:sequence>
      <xs:element name="queue" type="queueType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- resultPartialType -->
  <xs:complexType name="partialType">
    <xs:sequence>
      <xs:element name="action" type="actionType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- actionType -->
  <xs:complexType name="actionType">
    <xs:sequence>
      <xs:element name="queue" type="queueType" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="type" type="actionTypeContent" use="required"/>
  </xs:complexType>

  <!-- queueType -->
  <xs:complexType name="queueType">
    <xs:attribute name="queue_id" type="xs:string" use="required"/>
    <xs:attribute name="member" type="xs:boolean" use="required"/>
  </xs:complexType>

  <!-- actionTypeContent -->

```

```

<xs:simpleType name="actionTypeContent">
  <xs:restriction base="xs:string">
    <xs:enumeration value="added"/>
    <xs:enumeration value="removed"/>
    <xs:enumeration value="changed"/>
  </xs:restriction>
</xs:simpleType>
</xs:schema>

```

Примеры:**full:**

```

<?xml version="1.0"?>
<event name="queue_list_event" timestamp="1970/01/01 07:00:00">
<full>
  <queue queue_id="queue2" member="false"/>
  <queue queue_id="queue1" member="false"/>
</full>
</event>

```

partial:

```

<?xml version="1.0"?>
<event name="queue_list_event" timestamp="1970/01/01 07:00:00">
<partial>
  <action type="added">
    <queue queue_id="queue3" member="false"/>
  </action>
  <action type="removed">
    <queue queue_id="queue4" member="false"/>
  </action>
  <action type="changed">
    <queue queue_id="queue5" member="true"/>
  </action>
</partial>
</event>

```

queue_realtime_statistics_event.xsd

Событие со статистикой очереди в режиме реального времени.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>

  <xs:simpleType name="LimitType">
    <xs:union>
      <xs:simpleType>
        <xs:restriction base="xs:nonNegativeInteger"/>
      </xs:simpleType>
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="unbounded"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:union>
  </xs:simpleType>

  <xs:complexType name="QueueRealtimeStatisticsEventType">
    <xs:attribute name="queue_id" type="xs:string"/>
    <xs:attribute name="waiting_conversations_limit" type="LimitType"/>
    <xs:attribute name="waiting_conversations_c" type="xs:integer"/>
    <xs:attribute name="distributed_conversations_c" type="xs:integer"/>
  </xs:complexType>

  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:sequence>
            <xs:element name="queue_realtime_statistics" type="QueueRealtimeStatistic
sEventType"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

supervising_event.xsd

Нотификация о статусе супервизинга.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>
  <xs:include schemaLocation="conversation.xsd"/>

  <xs:simpleType name="SupervisingStatusType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="none"/>
      <xs:enumeration value="failed"/>
      <xs:enumeration value="connecting"/>
      <xs:enumeration value="consult"/>
      <xs:enumeration value="observing"/>
      <xs:enumeration value="conference"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:sequence>
            <xs:element name="supervising">
              <xs:complexType>
                <xs:attribute name="conversation_id" type="xs:string" use="option
al"/>
                <xs:attribute name="status" type="SupervisingStatusType"/>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Вспомогательные схемы HTTP API

Вспомогательные схемы используются в [СОБЫТИЯХ](#) или [КОМАНДАХ](#).

- [agent.xsd](#):
- [call_history_common.xsd](#):
- [conversation.xsd](#):
- [error.xsd](#):
- [ss.xsd](#):

agent.xsd:

Информация об операторе.


```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <!-- agentStatusType -->
  <xs:simpleType name="AgentStatusType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="logged_out"/>
      <xs:enumeration value="available"/>
      <xs:enumeration value="auxwork"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- agentActivityType -->
  <xs:simpleType name="AgentActivityType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="none"/>
      <xs:enumeration value="talking"/>
      <xs:enumeration value="acw"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="ChannelAvailability">
    <xs:restriction base="xs:string">
      <xs:enumeration value="busy"/>
      <xs:enumeration value="free"/>
      <xs:enumeration value="blocked"/>
      <xs:enumeration value="occupied"/>
      <xs:enumeration value="not_exists"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="Channel">
    <xs:sequence>
      <xs:element name="additional_info">
        <xs:complexType>
          <xs:anyAttribute/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="type" type="xs:string"/>
    <xs:attribute name="status" type="ChannelAvailability"/>
  </xs:complexType>

  <xs:complexType name="Channels">
    <xs:sequence>
      <xs:element name="channel" type="Channel" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

```

```
<xs:complexType name="AgentInfo">
  <xs:sequence>
    <xs:element name="channels" type="Channels"/>
  </xs:sequence>
  <xs:attribute name="agent_id" type="xs:string" use="optional"/>
  <xs:attribute name="display_name" type="xs:string" use="optional"/>
  <xs:attribute name="status" type="AgentStatusType" use="optional"/>
  <xs:attribute name="blocked" type="xs:boolean" use="optional"/>
  <xs:attribute name="activity" type="AgentActivityType" use="optional"/>
  <xs:attribute name="groups" type="xs:string" use="optional"/>
  <xs:attribute name="force_logout" type="xs:boolean" use="optional"/>
    <xs:attribute name="is_supervisor" type="xs:boolean" use="optional"/>
  </xs:complexType>

</xs:schema>
```

call_history_common.xsd:

Типы данных истории вызовов.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:simpleType name="SortType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="asc"/>
      <xs:enumeration value="desc"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="ActivatedBy">
    <xs:restriction base="xs:string">
      <xs:enumeration value="me"/>
      <xs:enumeration value="remote"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="UrlList">
    <xs:sequence>
      <xs:element name="url" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:simpleType name="PhoneNumber">
    <xs:restriction base="xs:string">
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="SSTransfer">
    <xs:attribute name="activated_by" type="ActivatedBy" use="required"/>
    <xs:attribute name="activator" type="PhoneNumber" use="required"/>
    <xs:attribute name="activator_display_name" type="xs:string" use="optional"/>
    <xs:attribute name="participant" type="PhoneNumber" use="required"/>
    <xs:attribute name="participant_display_name" type="xs:string" use="optional"/>
    <xs:attribute name="target" type="PhoneNumber" use="required"/>
    <xs:attribute name="target_display_name" type="xs:string" use="optional"/>
    <xs:attribute name="activate_time" type="xs:string" use="required"/> <!-- дата и время --
  >

    <xs:attribute name="call_record" type="xs:string" use="optional"/> <!-- url -->
    <xs:attribute name="is_master" type="xs:string" use="optional"/> <!-- только для истории
супервизора -->
  </xs:complexType>

  <xs:complexType name="SSList">
    <xs:sequence>
      <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="transfer" type="SSTransfer" minOccurs="0" maxOccurs="unbounded"
/>

```

```
        </xs:choice>  
    </xs:sequence>  
</xs:complexType>  
</xs:schema>
```

conversation.xsd:

Информация о разговоре.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <!--AutoCompleteTimeout-->
  <xs:simpleType name="AutoCompleteTimeout">
    <xs:restriction base="xs:nonNegativeInteger">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="3600"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="ConversationStatus">
    <xs:restriction base="xs:string">
      <xs:enumeration value="alerting"/>
      <xs:enumeration value="initiating"/>
      <xs:enumeration value="working"/>
      <xs:enumeration value="fax_receiving"/>
      <xs:enumeration value="fax_rejecting"/>
      <xs:enumeration value="hold"/>
      <xs:enumeration value="acw"/>
      <xs:enumeration value="released"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="ConversationType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="call"/>
      <xs:enumeration value="mail"/>
      <xs:enumeration value="chat"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="Direction">
    <xs:restriction base="xs:string">
      <xs:enumeration value="in"/>
      <xs:enumeration value="out"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="CCMode">
    <xs:restriction base="xs:string">
      <xs:enumeration value="normal"/>
      <xs:enumeration value="consult"/>
      <xs:enumeration value="consult_conference"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="Conversation">

```

```

<xs:sequence>
  <xs:element name="additional_info" minOccurs="0" maxOccurs="1">
    <xs:complexType>
      <xs:sequence>
        <xs:any namespace="##any" processContents="lax" minOccurs="0" maxOccurs="
unbounded" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:sequence>
<xs:attribute name="id" type="xs:string" use="required"/>
<xs:attribute name="type" type="ConversationType" use="required"/>
<xs:attribute name="status" type="ConversationStatus" use="required"/>
<xs:attribute name="start_time" type="xs:string" use="required"/>
<xs:attribute name="answer_time" type="xs:string" use="optional"/>
<xs:attribute name="release_time" type="xs:string" use="optional"/>
<xs:attribute name="fax_receiving_time" type="xs:string" use="optional"/>
<xs:attribute name="acd" type="xs:boolean" use="required"/>
<xs:attribute name="queue_id" type="xs:string" use="optional"/>
<xs:attribute name="queue_description" type="xs:string" use="optional"/>
<xs:attribute name="direction" type="Direction" use="required"/>
<xs:attribute name="display_name_a" type="xs:string" use="required"/>
<xs:attribute name="display_name_b" type="xs:string" use="required"/>
<xs:attribute name="digits_a" type="xs:string" use="optional"/>
<xs:attribute name="digits_b" type="xs:string" use="optional"/>
<xs:attribute name="fax_receive" type="xs:boolean" use="optional"/>
<xs:attribute name="ref" type="xs:string" use="required"/>
<xs:attribute name="auto_complete_launched" type="xs:boolean" use="required"/>
<xs:attribute name="auto_complete_timeout" type="AutoCompleteTimeout" use="optional"/>
<xs:attribute name="description" type="xs:string" use="optional"/>
<xs:attribute name="is_consult" type="xs:boolean" use="optional"/>
<xs:attribute name="workitem_id" type="xs:string" use="optional"/>
</xs:complexType>
</xs:schema>

```

error.xsd:

Информация об ошибке.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" id="error">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Error message for HttpTerminal application.
      Copyright 2015, Eltex. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <!-- <error> -->
  <xs:complexType name="errorType">
    <xs:attribute name="cmd" type="xs:token" use="required" />
    <xs:attribute name="reason" type="errorReasonType" use="required" />
    <xs:attribute name="body" type="xs:string" use="optional" />
    <!-- entity="[nonempty_string()]" -->
    <xs:attribute name="entity" type="xs:normalizedString" use="optional" />
  </xs:complexType>
  <!-- </error> -->

  <!-- Error reasons description -->
  <xs:simpleType name="errorReasonType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="unknown_error" />
      <xs:enumeration value="bad_token_info" />
      <xs:enumeration value="set_error" />
      <xs:enumeration value="exception" />
      <xs:enumeration value="error" />
      <xs:enumeration value="invalid_input_xml" />
      <xs:enumeration value="nocommand" />
      <xs:enumeration value="invalid_xml" />
      <xs:enumeration value="bad_licence" />
      <xs:enumeration value="profile_in_use" />
      <xs:enumeration value="no_rights" />
      <xs:enumeration value="parse_error" />
      <xs:enumeration value="exit" />
      <xs:enumeration value="canceled_by_user" />
      <xs:enumeration value="not_exists" />
      <xs:enumeration value="already_declared" />
      <xs:enumeration value="restricted_name"/>
      <xs:enumeration value="domain_not_exists"/>
      <xs:enumeration value="domain_not_found"/>
      <xs:enumeration value="profile_not_exists"/>
      <xs:enumeration value="profile_not_found"/>
      <xs:enumeration value="ivr_restrictions_profile_not_exists"/>
      <xs:enumeration value="badflags" />
      <xs:enumeration value="badargs" />
      <xs:enumeration value="badinput" />
      <xs:enumeration value="aborted" />
      <xs:enumeration value="no_enough_addresses" />
    </xs:restriction>
  </xs:simpleType>

```

```
<xs:enumeration value="no_enough_ifaces" />
<xs:enumeration value="invalid_range" />
<xs:enumeration value="const" />
<xs:enumeration value="db_error" />
<xs:enumeration value="invalid_value" />
<xs:enumeration value="name_already_in_use"/>
<xs:enumeration value="no_such_gateway"/>
<xs:enumeration value="bad_gateway_type"/>
<xs:enumeration value="no_signaling_gateway"/>
<xs:enumeration value="no_trunk_gateway"/>
<xs:enumeration value="no_such_opc"/>
<xs:enumeration value="no_such_dpc"/>
<xs:enumeration value="not_equal_dpc"/>
<xs:enumeration value="bad_internal_iface_format"/>
<xs:enumeration value="interface_already_exists"/>
<xs:enumeration value="port_already_in_use"/>
<xs:enumeration value="no_such_port"/>
<xs:enumeration value="session_not_found"/>
<xs:enumeration value="lock_not_found"/>
<xs:enumeration value="already_blocked"/>
<xs:enumeration value="bad_user_server_type"/>
<xs:enumeration value="already_exists"/>
<xs:enumeration value="cannot_get_data"/>
<xs:enumeration value="no_such_user"/>
<xs:enumeration value="connection_trouble"/>
<xs:enumeration value="name_is_busy"/>
<xs:enumeration value="context_not_found"/>
<xs:enumeration value="alias_declaration_error"/>
<xs:enumeration value="context_validation_error"/>
<xs:enumeration value="invalid_user_name"/>
<xs:enumeration value="invalid_alias_digits"/>
<xs:enumeration value="invalid_address_format"/>
<xs:enumeration value="invalid_routing_context"/>
<xs:enumeration value="no_such_group"/>
<xs:enumeration value="not_exists_agents"/>
<xs:enumeration value="zone_used_in_resource"/>
<xs:enumeration value="zone_not_found"/>
<xs:enumeration value="site_used_in_resource"/>
<xs:enumeration value="site_not_found"/>
<xs:enumeration value="bad_node"/>
<xs:enumeration value="undefined_properties"/>
<xs:enumeration value="ivr_scripts_domain_limit"/>
<xs:enumeration value="ivr_script_restricted_blocks"/>
<xs:enumeration value="ivr_script_too_many_blocks"/>
<xs:enumeration value="ivr_script_bad_first_block"/>
<xs:enumeration value="ivr_script_bad_pass"/>
<xs:enumeration value="already_defined"/>
<xs:enumeration value="connection_failed"/>
<xs:enumeration value="bad_id"/>
<xs:enumeration value="src_bridge_not_exists"/>
```



```

    <xs:enumeration value="src_bridge_domain_not_exists"/>
    <xs:enumeration value="dst_bridge_already_exists"/>
    <xs:enumeration value="without_value"/>
    <xs:enumeration value="domain_limit"/>
    <xs:enumeration value="licence_limit"/>
    <!-- More reasons can be added here -->
  </xs:restriction>
</xs:simpleType>
<!-- End of error reasons description -->

</xs:schema>

```

ss.xsd:

Информация об активированных в разговоре услугах.

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="SSTransfer">
    <xs:attribute name="activator" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="ru">
          Участник, активировавший трансфер
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="participant" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="ru">
          Этот участник изначально разговаривал с activator, и в конечном итоге стал
          разговаривать с target
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="target" type="xs:string" use="required">
      <xs:annotation>
        <xs:documentation xml:lang="ru">
          Участник на которого был осуществлен трансфер
        </xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="activate_time" type="xs:string" use="required"/>
  </xs:complexType>

</xs:schema>

```

Pubsub API

Pubsub API — специальный вариант API, который позволяет создавать сессии, не привязанные к конкретному оператору. Вместо этого оно позволяет доверенным клиентам (web-приложениям) получать некоторую информацию о Call-центра, используя учетные данные созданные для клиента, а не для оператора. Цель данного API — снижение нагрузки на отдельные компоненты SSW посредством вынесения части логики подписок (рассылка уведомлений конечным получателям или агрегация событий и отложенная их отправка) на доверенный клиент. Подчеркивается, что это должен быть именно доверенный клиент, так как данные подписки, при неправильном их использовании могут создавать избыточную нагрузку на SSW, мешая ему выполнять функции обработки вызовов.

- [Команды Pubsub API](#)
- [События Pubsub API](#)

URL-пути имеют вид /service/cc/pubsub/*

В текущий момент времени передача данных между клиентом и сервером (SSW) будет осуществляться по вебсокетах. AMQP возможно был бы более предпочтительным вариантом интеграции, но большинство популярных клиентов AMQP не поддерживают используемую нами версию AMQP (0.10).

Команды Pubsub API

Команды могут выполняться либо как отдельный HTTP запрос, либо по вебсокетах.

Полный список команд:

- login
- logout (см. команды HTTP API выше)
- monitor (см. команды HTTP API выше)

pubsub_login.xsd:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="Request">
    <xs:attribute name="login" type="xs:string" use="required"/>
    <xs:attribute name="password_hash" type="xs:base64Binary" use="required"/>
    <xs:attribute name="password_salt" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="Response">
    <xs:all>
      <xs:element name="ok"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="Request"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="Response"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

События Pubsub API

Полный список событий:

- line_status_event

line_status_event.xsd:

Статус линии абонента SSW

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>
  <xs:include schemaLocation="conversation.xsd"/>

  <xs:simpleType name="LineStatusType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="idle"/>
      <xs:enumeration value="alerting"/>
      <xs:enumeration value="talking"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:all>
            <xs:element name="line">
              <xs:complexType>
                <xs:attribute name="status" type="LineStatusType"/>
              </xs:complexType>
            </xs:element>
          </xs:all>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

Подключение к ecss-cc-ui-api

Для того чтобы иметь возможность взаимодействовать с [API колл-центра](#) необходимо установить проху-сервер **ecss-cc-ui**.

Установка производится командой:

```
sudo apt install ecss-cc-ui
```

После установки актуальные настройки можно проверить в файле **/usr/share/ecss/ecss-cc-ui-api/dist/websockets/src/config-production.js**:

```
/usr/share/ecss/ecss-cc-ui-api/dist/websockets/src/config-production.js
exports.LOG_FILENAME = '/var/log/ecss/cc-ui/websocket.log';
exports.LOG_LEVEL = 'error';
exports.WEBSOCKET_HOST = '0.0.0.0';
exports.WEBSOCKET_PORT = 8091;
exports.USE_SSL = true;
exports.SSL_CERT_CRT = '/etc/ecss/ssl/ecss10.crt';
exports.SSL_CERT_KEY = '/etc/ecss/ssl/ecss10.key';
```


Там же можно установить уровень логирования (error, info, debug), изменить порт, отключить SSL (по умолчанию включен).


После изменения настроек необходимо перезапустить сервис:

```
sudo service ecss-cc-ui-api restart
```

Так же ecss-cc-ui позволяет использовать профили подключения, настраиваемые в файле **/etc/ecss/ecss-cc-ui-api/config.yaml**:

```
default:
  ecss:
    host: 10.25.64.21
    port: 8086
    domain:
    user: pubsub
    password: pubsub
    redmine: null
    ldap_integration_auth: false
  sql:
    host: 10.25.64.21
    port: 3306
    database: ecss_address_book
    user: root
    password: root
    external: mvd
  custom:
    brand: default
```

 После изменения потребуется перезапуск `ecss-cc-ui-api.service`, чтобы изменения вступили в силу.

 **ECSS Call Center API** работает на протоколе [websockets](#). Обмен происходит только в текстовом виде. Запросы оформляются в виде JSON.

Использование технологии websocket

Для того чтобы иметь возможность взаимодействовать с [API колл-центра](#), необходимо открыть websocket-соединение до проху-сервера.

Проху-сервер это nodejs-приложение которое доступно после установки пакета [ecss-cc-ui](#), доступно по [wss://<ecss-cc-ui_ip>:<ecss-cc-ui_port>](#).

Порт задается при установке пакета [ecss-cc-ui](#), по умолчанию 8091.

Например, используя реализацию [rxjs/webSocket](#), это можно сделать так:

```
this.ws = websocket(this.url)
this.ws.subscribe(
  (response: EltexResponse<any>) => this.fullDataStream.next(response),
  () => this._reconnectStream.next(),
  () => this._reconnectStream.next(),
)
```

Рсар такого вызова выглядит следующим образом:

```
GET / HTTP/1.1
Host: 192.168.116.130:8091
Connection: Upgrade
Pragma: no-cache
Cache-Control: no-cache
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
Upgrade: websocket
Origin: 192.168.116.130:4200
Sec-WebSocket-Version: 13
Accept-Encoding: gzip, deflate, br
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: csrftoken=use5JWejQXZB1tx9LYKfsZzOXOHTnbs0Ytkc0BxQZDIbkyn7BRdC6SHxp0GH2Vey; JSESSIONID=node0zeowgwu8yiu618sgj0akvjq771.node0; wcf_sid=quvcn3kzptuaqd8sign109uuh0mul5ec
Sec-WebSocket-Key: A6Ljrs3D5xFKKQcym5R9Yw==
Sec-WebSocket-Extensions: permmessage-deflate; client_max_window_bits

HTTP/1.1 101 Switching Protocols
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Accept: /8tybArQZvyYICIM+33XZ37yw/Rw=
```

Протокол общения

Существует два типа входящих запросов

1. Запрос-ответ

2. Запрос-ответ для подписки на уведомления + после такого запроса приходят уведомления по подписке

Входящие запросы

```
interface EltexRequest {  
    /**  
     * Имя команды (url).  
     */  
    action: string  
    /**  
     * Идентификатор запроса, должен быть уникальным значением, равен или  
    больше нуля.  
     */  
    requestId: number  
    /**  
     * Дополнительные параметры запроса  
     */  
    payload: object  
}
```

Ответы на запросы

```
interface EltexResponse<T> {  
    /**  
     * Имя команды, идентично запросу, либо неизвестное ранее значение для  
    уведомлений или критических ошибок  
     */  
    action: string  
    /**  
     * Идентификатор запроса, либо 0 для уведомлений  
     */  
    requestId: number  
    /**  
     * Полезная нагрузка, объект, строка, или null  
     */  
    payload: T  
    /**  
     * Статус: 200 OK, 401 неверные авторизационные данные, 500 для ошибок и  
    403 если сервер закрыл сессию пользователя  
     */  
    status: number  
}
```

Для большинства запросов требуется предварительная авторизация.

Команды не требующие авторизации:

[login](#), [auth](#), [profile](#).

Уведомления по подписке присылаются с `requestId = 0`.

Параметры для запроса передаются в `payload` в виде вложенных объектов.

- ✔ Сразу после [login](#) `ecss-cc-ui-api` открывает `websocket` для получения данных от сервера и передачи команд от пользователя.

```
GET /dom1/service/cc/arm/websockets HTTP/1.1
Sec-WebSocket-Version: 13
Sec-WebSocket-Key: wP+F1PkzHDC0bA/mzkrkFQ==
Connection: Upgrade
Upgrade: websocket
Cookie: token=064803c90ec3b72b; Version=1; Path=/
Sec-WebSocket-Extensions: permessage-deflate; client_max_window_bits
Host: 10.25.64.21:8086

HTTP/1.1 101 Switching Protocols
connection: Upgrade
upgrade: websocket
sec-websocket-accept: w3h/LIiEKYiZDyFqIiH2Lk8HEHI=
```


ecss-cc-ui API

Список команд:

Команда auth

Параметры

AuthRequest

```
{
  /**
   * Токен сессии, возвращаемый в ответ на команду login
   */
  token: string
}
```

Коды ответов:

200 — в случае успеха;

403 — в случае устаревания токена и закрытия сессии

Пример:

Запрос:

```
{"action": "auth", "requestId": 1, "payload": {"token": "fa6ec020bfc8236e377960143579c706:da005b18370faeff3a8337961a32d759da187e08f5247993719ef88872ea8dff8fbc31141dc8cabe33ce4dc73b4c0866"}}
```

Код ответа:

403

Ответ:

```
{"action": "auth", "requestId": 1, "payload": {}, "status": 403}
```

❗ Серверу команда `auth` не посылается в явном виде, `ecss-cc-ui-api` проверяет по списку активных сессий есть ли запрашиваемый токен или нет среди них.

✔️ Повторная аутентификация по токenu возможна в случае неполадок на сети, если ни пользователь не нажимал явно `logout`, не сервер не закрывал сессию пользователя принудительно.

Время жизни токена не ограничено, пул открытых токенов подчищается если не вызывая `logout` например просто закрыть вкладку браузера (через 7 секунд). Также токен считается "подозрительно устаревшим", если не удалось передать ответ от сервера клиенту.

Команда `call/accept`

Команда ответа на входящий разговор.

Параметры запроса

```
{  
  // Идентификатор вызова  
  callId: string  
}
```

✔ callId: string может быть получен из уведомления по подписке [call/subscribe](#) (id: string)

Параметры ответа

```
{  
  ok: {}  
}
```

Коды ответов:

200

Пример:

Запрос:

```
{  
  "action": "call/accept",  
  "requestId": 66,  
  "payload": {  
    "callId": "064805dde9d56f28"  
  }  
}
```

Код ответа:

200

Ответ:

```
{
  "action": "call/accept",
  "requestId": 66,
  "payload": {
    "ok": {}
  },
  "status": 200
}
```

Команда call/complete

Команда завершения пост-процессинга указанного разговора.

Параметры запроса

```
{
  // Идентификатор вызова
  callId: string
}
```

✔ callId: string может быть получен из уведомления по подписке [call/subscribe](#) (id: string)

Параметры ответа

```
{
  ok: {}
}
```

Коды ответов:

200

Пример:

Запрос:

```
{
  "action": "call/complete",
  "requestId": 100,
  "payload": {
    "callId": "064806432b729693"
  }
}
```

Код ответа:

200

Ответ:

```
{
  "action": "call/complete",
  "requestId": 100,
  "payload": {
    "ok": {}
  },
  "status": 200
}
```

Команда call/getCallRecord

Получение URL-записи разговора на этапе пост-обработки вызова


Параметры запроса

```
{
  // Ссылка на диалог вызова, не путать с идентификатором вызова (id)
  callRef: string
}
```

- ✓ callRef: string может быть получен из уведомления по подписке [call/subscribe](#) (callRef?: string)

Параметры ответа

```
{
  url: string
}
```

 Если запись вызова отключена, url будет пустой строкой.

Коды ответов:

200

Пример:

Запрос:


```
{
  "action": "call/getCallRecord",
  "requestId": 39,
  "payload": {
    "callRef": "7077745701865141859"
  }
}
```

Код ответа:

200

Ответ:

```
{
  "action": "call/getCallRecord",
  "requestId": 39,
  "payload": {
    "url": "http://10.25.64.21:9990/domain/dom1/records/
2020_12_16/2020-12-16_14-07-40_t_6601-2004.wav"
  },
  "status": 200
}
```

 В данном примере 10.25.64.21:9990 адрес где установлен ecss-restfs, для получения записи необходимо выполнить соответствующий http-запрос.

Команда call/makeCall

Сделать вызов на номер или по идентификатору оператора.

Параметры запроса

```
{
  // Номер для вызова, может быть не задан, если указан operatorId
  number?: string,
  // Идентификатор оператора, может быть не задан, если указан номер
  operatorId?: string
}
```

Параметры ответа

```
{
  "ok": {
    // Идентификатор вызова
    internalCallId?: string
  }
}
```

Коды ответов:

200 — в случае успеха;

403 — оператор, который пытается выполнить команду, не зарегистрирован на телефонном аппарате.

409 — эта команда уже выполняется для данного оператора.

Пример:

Запрос:

```
{
  "action": "call/makeCall",
  "requestId": 14,
  "payload": {
    "number": "2003"
  }
}
```

Код ответа:

200

Ответ:

```
{
  "action": "call/makeCall",
  "requestId": 14,
  "payload": {
    "ok": {
      "internalCallId": "0648051f8980f32c"
    }
  },
  "status": 200
}
```

Код ответа:

500

Ответ:

```
{
  "action": "call/makeCall",
  "requestId": 50,
  "payload": "{\"cmd\": \"make_call\", \"reason\": \"cannot_perform_cmd\", \"body\": \"timeout\"}",
  "status": 500
}
```

Вызов по идентификатору оператора

Пример:

Запрос:

```
{
  "action": "call/makeCall",
  "requestId": 30,
  "payload": {
    "operatorId": "2"
  }
}
```

Код ответа:

200

Ответ:


```
{
  "action": "call/makeCall",
  "requestId": 30,
  "payload": {
    "ok": {
      "internalCallId": "06480558b348fe6f"
    }
  },
  "status": 200
}
```

Команда call/reject

Команда для отклонения указанного разговора.

Параметры запроса

```
{
  // Идентификатор вызова
  callId: string
}
```

- ✔ callId: string может быть получен из уведомления по подписке [call/subscribe](#) (id: string)

Параметры ответа

```
{
  ok: {}
}
```

Коды ответов:

200

Пример:

Запрос:

```
{
  "action": "call/reject",
  "requestId": 82,
  "payload": {
    "callId": "0648061942208b2c"
  }
}
```

Код ответа:

200

Ответ:

```
{
  "action": "call/reject",
  "requestId": 82,
  "payload": {
    "ok": {}
  },
  "status": 200
}
```

Команда call/subscribe

Подписка на активные вызовы оператора

Параметры запроса

```
{
  operatorIds: string[]
}
```

Параметры ответа

```
CallFromServer[]
```

```

export interface CallFromServer {
  acd: 'true' | 'false'
  answerTime: iso8601
  numberA: string
  numberB: string
  direction: string
  displayNameA: string
  displayNameB: string
  // Ссылка на диалог вызова, используется для получения записи разговора на этапе
постобработки
  callRef?: string
  conferenceId?: string
  faxReceive: 'true' | 'false'
  faxUrls: Url[]
  // Идентификатор вызова, используется для управления вызовом: принять, отбить, завершить и
др.
  id: string
  onConsultation?: 'true' | 'false'
  operatorId: string
  isConsult: 'true' | 'false'
  queueDescription: string
  queueId: string
  // Служебная ссылка на диалог, не используется
  ref: string
  releaseTime: iso8601
  startTime: iso8601
  status: ActiveCallStatus
  supervising?: InterceptModes
  type: ActiveCallType
}

type Url = string

export enum ActiveCallStatus {
  working = 'working',
  alerting = 'alerting',
  initiating = 'initiating',
  fax_receiving = 'fax_receiving',
  fax_rejecting = 'fax_rejecting',
  hold = 'hold',
  acw = 'acw',
  released = 'released',
  finished = 'finished',
}

export enum ActiveCallType {
  call = 'call',
  mail = 'mail',
  chat = 'chat',
}

```

```

}

export enum InterceptModes {
  observing = 'observing',
  consult = 'consult',
  conference = 'conference',
}

```

✔ Идентификатор вызова `id: string` используется в командах `call/accept`, `call/reject`, `call/complete`

✔ Ссылка на диалог вызова `callRef?: string` используется в команде `call/getCallRecord`

Коды ответов:

200

Пример:

Запрос:

```

{
  "action": "call/subscribe",
  "requestId": 13,
  "payload": {
    "operatorIds": [
      "4"
    ]
  }
}

```

Код ответа:

200

Ответ:

```

{
  "action": "call/subscribe",
  "requestId": 13,
  "payload": {},
  "status": 200
}

```

Сразу после этого приходит уведомление с пустым списком вызовов

```
{
  "action": "event",
  "requestId": 0,
  "payload": {
    "event": "call/subscribe",
    "payload": []
  },
  "status": 200
}
```

При входящем вызове, например с номера 6601 приходит уведомление

```
{
  "action": "event",
  "requestId": 0,
  "payload": {
    "event": "call/subscribe",
    "payload": [
      {
        "id": "0648139dc0cb2f00",
        "type": "call",
        "status": "alerting",
        "acd": "false",
        "direction": "in",
        "ref": "{<<\0648139dbe8a77af\>>,95}",
        "intervention": "true",
        "callRef": "1993245726",
        "isConsult": false,
        "displayNameA": "Матвей Владимирович",
        "displayNameB": "Константинопольский Константин",
        "alertingTime": "2020/12/16 07:59:08",
        "faxReceive": "true",
        "startTime": "2020-12-16T07:59:08.000+07:00",
        "numberA": "6601",
        "numberB": "2004",
        "operatorId": "4",
        "faxUrls": []
      }
    ]
  },
  "status": 200
}
```

✓ Для того чтобы принять вызов, нужно отправить команду `call/accept`, используя id вызова

Команда login

Параметры

interface Login

```
{  
  /**  
   * Логин оператора  
   */  
  login: string  
  /**  
   * Пароль оператора  
   */  
  password: string  
  /**  
   * Номер телефона оператора  
   */  
  number: string  
  /**  
   * Имя профиля настроек сервера  
   */  
  profile: string  
  /**  
   * Домен, может быть не указан, если задан в настройках профиля  
   */  
  domain?: string  
}
```

ОТВЕТ

interface AuthResponse

```

{
    /**
     * Токен авторизации
     */
    token: string
    /**
     * Признак, является ли оператор супервизором
     */
    supervisor: boolean
    /**
     * Дополнительные возможности оператора
     */
    capabilities: UserCapabilities
    /**
     * Время, после которого пользователь будет разлогинен, если он не выполняет никаких действий
     */
    autoLogoutTimeout: number|null
}

UserCapabilities {
    /**
     * Признак, что пользователь видит информацию о вызовах в очереди
     */
    operatorGetInQueueCalls: boolean
}

```

Коды ответов:

200 — в случае успеха;

401 — Текстовые сообщения:

- Requested number is occupied — если номер уже занят другим оператором
- Requested number is not exist — если номер не существует
- Invalid login or password — некорректно указан логин или пароль
- System error. Abonent manager is unavailable — системная ошибка
- Service is not activated for this phone number — услуга Агент КЦ не активирована для выбранного номера
- timeout — при несуществующем домене
- domain not exists — при несуществующем домене
- Gateway timeout — при недоступности сервера

Пример:

Запрос:

<http://192.168.116.130:8091>

```
{
  "action": "login",
  "requestId": 16,
  "payload": {
    "login": "1",
    "password": "1111",
    "profile": "default",
    "number": "2004",
    "domain": "arko",
    "role": "operator"
  }
}
```

Код ответа:

200

Ответ:

```
{
  "action": "login",
  "requestId": 16,
  "payload": {
    "token":
"6601c8d19059c9b20787ec7e95f2d9d6:986d6fd21c3af83df479947f879e476ec4bb180e69c8307230828cfa5eb91b0
978d419763917579a54d5f89dcb2f0936",
    "capabilities": {
      "operatorGetInQueueCalls": true
    },
    "autoLogoutTimeout": 5,
    "agentId": "1",
    "supervisor": false
  },
  "status": 200
}
```

- ✔ Сразу после login `ecss-cc-ui-api` открывает `websocket` для получения данных от сервера и передачи команд от пользователя.

Команда logout

Команда выхода из Call-центра.

Параметры запроса

LogoutRequest

```
{  
  /**  
   * Индикатор что логатут произошел автоматически, а не по явному действию пользователя  
   */  
  auto?: boolean  
}
```

Коды ответов:

200

403

Пример:

Запрос:

```
{  
  "action": "logout",  
  "requestId": 29,  
  "payload": {  
    "auto": false  
  }  
}
```

Код ответа:

403

Ответ:

```
{  
  "action": "logout",  
  "requestId": 0,  
  "payload": {},  
  "status": 403  
}
```

Команда profiles

Возвращает список [профилей](#) прописанных в `/etc/ecss/ecss-cc-ui-api/config.yaml`. Если в профиле задан домен, то именно он будет использоваться при [подключении](#) к ECSS, если нет — его необходимо указать

Параметры

```
{
  null
}
```

Ответ

```
Profile {
/**
 * Имя профиля
 */
name: string
/**
 * Признак что имя профиля указано в настройках
 */
domain: boolean
/**
 * Адрес страницы для интеграции с redmine
 */
redmine: string | null
/**
 * Признак, активирована ли авторизация через LDAP
 */
ldap_integration_auth: boolean
/**
 * Имя профиля брендирования для веб-интерфейса
 */
brand: string
}
Profile[]
```

Коды ответов:

200

Пример:

Запрос:

```
{
  "action": "profiles",
  "requestId": 1,
  "payload": null
}
```

Код ответа:

200

Ответ:

```
{
  "action": "profiles",
  "requestId": 1,
  "payload": [
    {
      "name": "default",
      "redmine": null,
      "ldap_integration_auth": false,
      "domain": false,
      "brand": "default"
    }
  ],
  "status": 200
}
```

✔ Настройки профиля задаются при установке пакета `ecss-cc-ui`

Call API

Подразделы

Call API представляет собой набор http-команд, event-ов, используя которые можно управлять вызовами в рамках ECSS-10.

Call API поддерживает 2 типа контента:

1. XML (см. схемы приведенные в разделе "XSD файлы")
2. DOM закодированный в JSON.

Предположим, XML имеет следующую обобщенную структуру:

```
<ELEMENT_1 ATTRIBUTE_1=VALUE_1, ... ATTRIBUTE_N=VALUE_N>
... CONTENT
</ELEMENT_1>
...
<ELEMENT_N ...>
... CONTENT
</ELEMENT_N>
```

CONTENT рекурсивно определяет такую же структуру для всех вложенных узлов.

Данная структура представляется в виде JSON следующего вида:

```
[
  {
    "_name": "ELEMENT_1",
    "_attributes": {"ATTRIBUTE_1": "VALUE_1", ... "ATTRIBUTE_N": "VALUE_N"},
    "_content": CONTENT
  },
  ...
  {
    "_name": "ELEMENT_N",
    "_content": CONTENT
  }
]
```

Пример

Возьмем следующую структуру xml:

```
<a>
  <b x="test">
</a>
<c/>
```

Она будет преобразована в следующую структуру JSON:

```
[
  {
    "_name": "a",
    "_content": [
      {
        "_name": "b",
        "_attributes": {"x": "test"}
      }
    ]
  },
  {
    "_name": "c"
  }
]
```

⚠️ Ограничения

Со стороны JSON поддерживаются только значения типа "строка". Это значит, что если по API подразумевается передача численного аргумента, или аргумента типа `boolean`, оно должно быть представлено в виде строки:> {

```
> {
> "value": 42
> }
>
```

Должно выглядеть следующим образом:

```
> {
> "value": "42"
> }
>
```

- Со стороны XML не поддерживаются неймспейсы и текстовый контент.

Call HTTP API

Кроме HTTP ответов, указанных для той или иной команды, сервер использует следующие коды ошибок независимо от команды:

- 403 с текстовым содержимым "Permission denied" — у пользователя нет прав для выполнения данного запроса;
- 503 — команда выполняется слишком долго;
- 500 — ошибка парсинга запроса и другие внутренние ошибки сервера;
- Другие HTTP коды.

Ошибка 403 с текстом Permission denied, в отличие от других ошибок не должна возникать в правильно реализованном клиенте. Правильно реализованный клиент может (и должен) узнавать список возможностей и не предлагать пользователю выполнять действия, которые запрещены политикой доступа (см. Capabilities в Команда login и раздел Специальные Разрешения).

Команды HTTP Call API

Выполнить то или иное действие. Команды могут выполняться либо как отдельный HTTP запрос, либо по вебсокетах.

Полный список команд:

HTTP Call API - answer.xsd

Ответить на входящий вызов.

Шаблон HTTP-запроса:

<http://host:port/<domain>/service/cc/arm/answer>

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="Answer">
    <xs:attribute name="conversation_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="answer" type="Answer"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответы:

200

404 - Нет такого вызова

HTTP Call API - get_commands_list.xsd

Команда просмотра списка API команд.

Шаблон HTTP-запроса:

http://host:port/<domain>/service/cc/arm/get_commands_list

XSD-схема XML-запроса/ответа:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- requestType -->
  <xs:complexType name="commandType">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="description" type="xs:string" use="required"/>
    <xs:attribute name="xsd" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:element name="commands">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="command" type="commandType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Ответы:

200

HTTP Call API - login.xsd

Аутентификация в Call API.

Шаблон HTTP-запроса:

<http://host:port/<domain>/service/cc/arm/login>

XSD-схема XML-запроса/ответа:


```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="Request">
    <xs:attribute name="client_id" type="xs:string" use="required"/>
    <xs:attribute name="api_key" type="xs:string" use="required"/>
    <xs:attribute name="phone_number" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="Response">
    <xs:all>
      <xs:element name="ok"/>
    </xs:all>
  </xs:complexType>
  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request" type="Request"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:all>
        <xs:element name="response" type="Response"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответы:

200

403 - Текстовые сообщения:

- Invalid credentials
- *И некоторые другие, не определенные строго сообщения...*

HTTP Call API - logout.xsd

Завершение текущей сессии Call API.

Тело запроса отсутствует.

Ответы:

200

HTTP Call API - make_call.xsd

Сделать исходящий вызов.

Шаблон HTTP-запроса:

http://host:port/<domain>/service/cc/arm/make_call

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="MakeCall">
    <xs:attribute name="to_number" type="xs:string" use="required"/>
    <!--From number does not required if it specified in login command-->
    <xs:attribute name="from_number" type="xs:string" use="optional"/>
    <xs:attribute name="subject_of_call" type="xs:string" use="optional"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="make_call" type="MakeCall"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответы:

200

HTTP Call API - monitor.xsd

Команда для подписки на событие.

Шаблон HTTP-запроса:

<http://host:port/<domain>/service/cc/arm/monitor>

XSD-схема XML-запроса/ответа:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- baseEventType -->
  <xs:complexType name="baseEventType" abstract="true">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="entity" type="xs:string" use="optional"/>
    <xs:attribute name="timestamp" type="xs:string" use="optional"/>
  </xs:complexType>
  <!-- monitorEventType -->
  <xs:complexType name="monitorEventType">
    <xs:complexContent>
      <xs:extension base="baseEventType"/>
    </xs:complexContent>
  </xs:complexType>
  <!-- startType -->
  <xs:complexType name="startType">
    <xs:sequence>
      <xs:element name="event" type="monitorEventType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- stopType -->
  <xs:complexType name="stopType">
    <xs:sequence>
      <xs:element name="event" type="monitorEventType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!-- monitor -->
  <xs:element name="monitor">
    <xs:complexType>
      <xs:all>
        <xs:element name="start" type="startType" minOccurs="0"/>
        <xs:element name="stop" type="stopType" minOccurs="0"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Ответы:

200

404 - Подписка на несуществующее событие

HTTP Call API - ping.xsd

Команда проверки активности текущей сессии. Продлить время жизни текущей сессии.

Шаблон HTTP-запроса:

<http://host:port/<domain>/service/cc/arm/ping>

XSD-схема XML-запроса/ответа:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Ответы:

200 - pong

HTTP Call API - reject.xsd

ОТКЛОНИТЬ ВЫЗОВ.

Шаблон HTTP-запроса:

<http://host:port/<domain>/service/cc/arm/reject>

XSD-схема XML-запроса/ответа:

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="error.xsd"/>

  <xs:complexType name="Reject">
    <xs:attribute name="conversation_id" use="required"/>
  </xs:complexType>

  <!-- in -->
  <xs:element name="in">
    <xs:complexType>
      <xs:all>
        <xs:element name="request">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="reject" type="Reject"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <!-- out -->
  <xs:element name="out">
    <xs:complexType>
      <xs:choice>
        <xs:element name="response">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ok"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="error" type="errorType"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Ответы:

200

404 - Нет такого вызова

События HTTP Call API

После подписки на то или иное событие с помощью команды monitor, по вебсокетам начинают приходить нотификации.

Полный список событий:

HTTP Call API - conversations_event.xsd

Информация о вызовах.

Шаблон HTTP-запроса:

http://host:port/<domain>/service/cc/arm/conversations_event

XSD-схема XML-запроса/ответа:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

  <xs:include schemaLocation="monitor.xsd"/>

  <xs:simpleType name="Status">
    <xs:restriction base="xs:string">
      <xs:enumeration value="alerting"/>
      <xs:enumeration value="talking"/>
      <xs:enumeration value="released"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="Direction">
    <xs:restriction base="xs:string">
      <xs:enumeration value="in"/>
      <xs:enumeration value="out"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="Conversation">
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="status" type="Status" use="required"/>
    <xs:attribute name="direction" type="Direction" use="required"/>
    <xs:attribute name="digits" type="xs:string" use="optional"/>
    <xs:attribute name="display_name" type="xs:string" use="optional"/>
    <xs:attribute name="remote_digits" type="xs:string" use="optional"/>
    <xs:attribute name="remote_display_name" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="ConversationsType">
    <xs:sequence>
      <xs:element name="conversation" type="Conversation" minOccurs="0" maxOccurs="unbounde
d"/>
    </xs:sequence>
  </xs:complexType>

  <xs:element name="event">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="baseEventType">
          <xs:sequence>
            <xs:element name="conversations" type="ConversationsType"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

```



```
</xs:schema>
```

API сервиса Автосекретарь

Добавление слов в базу данных

- Таблица full names
 - Добавление полных имен:
 - Удаление полных имен:
- Таблица aliases
 - Добавление алиаса:
 - Удаление алиаса:
- Таблица hard_aliases (жесткие алиасы)
 - Добавление
 - Удаление
- Таблица extended_word
 - Добавление
 - Удаление

i Данную документацию можно найти по адресу: <IP_HOST>:9010/api/v1/swagger где <IP_HOST> это ip адрес хоста с установленным пакетом ecss-clerk

Таблица full names

Описание: Таблица предназначена для добавления полных имен. После создания имени ему будет присвоен id, необходимый при добавлении алиаса.

Добавление полных имен:

Path: /api/v1/fullNames

Request-type: POST

Content-type: application/json

Требуемые поля в json: name

Запрос:

```
curl -X POST "<IP_HOST>:9010/api/v1/fullNames" -H "accept: application/json" -H "content-type: application/json" -d '{"name": "александр"}'
```

Ответ:

```
{
  "data": {
    "attributes": {
      "name": "александр"
    },
    "id": "1592",
    "type": "fullNames"
  },
  "jsonapi": {
    "version": "1.0"
  }
}
```

- IP_HOST - ip хоста

Удаление полных имен:

Path: /api/v1/fullNames

Request-type: DELETE

Content-type: application/json

Требуемые поля в json: Full Name ID

Пример:

Запрос:

```
curl -X DELETE "<IP_HOST>:9010/api/v1/fullNames/<Full_Name_ID>" -H "accept: application/json"
```

- IP_HOST - ip хоста
- Full_Name_ID - id присвоенный при создании полного имени

Таблица aliases

Описание: Таблица предназначена для добавления сокращений к полным именам (саша -> александр). Соответственно, fullName это id полного имени полученного из запроса FullNames. Связка alias => fullName должна быть уникальной, при попытке добавления дублирующийся записи будет возвращена ошибка.

Добавление алиаса:

Path: /api/v1/aliases

Request-type: POST

Content-type: application/json

Требуемые поля в json: alias и fullNameId

Запрос:

```
curl -x POST "<IP_HOST>:9010/api/v1/aliases" -H "accept: application/json" -H "content-type: application/json; charset=utf-8" -d '{"alias": "слава", "fullNameId": "83"}'
```

Ответ:

```
{
  "data": {
    "attributes": {
      "alias": "слава",
      "default": false
    },
    "id": "6451",
    "relationships": {
      "fullName": {
        "data": {
          "id": "83",
          "type": "fullNames"
        }
      }
    }
  },
  "type": "aliases"
},
"included": [
  {
    "attributes": {
      "name": null
    },
    "id": "83",
    "type": "fullNames"
  }
],
"jsonapi": {
  "version": "1.0"
}
}
```

- IP_HOST - ip хоста;
- fullNameId - id полного имени которое ставится в соответствии этому алиасу.

Удаление алиаса:

Path: /api/v1/aliases

Request-type: DELETE

Content-type: application/json

Требуемые поля в json: Alias ID

Пример:

```
Запрос:
curl -X DELETE "<IP_HOST>:9010/api/v1/aliases/<Alias_ID>" -H "accept: application/json"
```

- IP_HOST - ip хоста;
- Alias_ID - id присвоенный при создании алиаса.

Таблица hard_aliases (жесткие алиасы)

Описание: Жесткие алиасы предназначены для корректировки работы инструмента распознавания речи. К примеру, некоторое имя (целиком) может стабильно распознаваться как одна и та же фраза.

Эта таблица призвана решить данную проблему путем добавления фразы и определения для неё имени из телефонной книги.

Пример: На предприятии плохо распознается имя «игорь николаев», результатом распознавания часто является «игорь николай». Для решения этой проблемы нужно добавить запись в таблицу вида «игорь николай» -> «игорь николаев».

Поле alias должно быть уникальным, иначе при добавлении вернется ошибка.

Добавление

Path: /api/v1/hardAliases

Request-type: POST

Content-type: application/json

Требуемые поля в json: alias и realName

```
Запрос:
curl -X POST "<IP_HOST>:9010/api/v1/hardAliases" -H "accept: application/json" -H "content-type: application/json" -d '{"alias": "игорь николай", "realName": "игорь николаев"}'
```

```
Ответ:
{
  "data": {
    "attributes": {
      "alias": "игорь николай",
      "realName": "игорь николаев"
    },
    "id": "34",
    "type": "hardAliases"
  },
  "jsonapi": {
    "version": "1.0"
  }
}
```

- IP_HOST - ip хоста.

Удаление

Path: /api/v1/hardAliases

Request-type: DELETE

Content-type: application/json

Требуемые поля в json: Hard_alias_ID

Пример:

Запрос:

```
curl -X DELETE "<IP_HOST>:9010/api/v1/hardAliases/<Hard_alias_ID>" -H "accept: application/json"
```

- IP_HOST - ip хоста;
- Hard_alias_ID - id hard aliases которого нужно удалить.

Таблица extended_word

Основные поля: word

Описание: Таблица нужна для компиляции модели распознавания речи. Корпус новых слов, которые будут добавлены в модель строятся из двух источников: телефонной книги и данной таблицы.

Если по какой-то причине требуется распознавание слов, отсутствующих в телефонной книге, то вам поможет эта таблица.

Добавление

Path: /api/v1/words

Request-type: POST

Content-type: application/json

Требуемые поля в json: words

Пример:

```
Запрос:
curl -X POST "<IP_HOST>:9010/api/v1/words" -H "accept: application/json" -H "content-type: application/json" -d '{"word": "элтекс"}'
```

```
Ответ:
{
  "data": {
    "attributes": {
      "word": "элтекс"
    },
    "id": "7",
    "type": "words"
  },
  "jsonapi": {
    "version": "1.0"
  }
}
```

- IP_HOST - ip хоста;

Удаление

Path: /api/v1/words

Request-type: DELETE

Content-type: application/json

Требуемые поля в json: Word_ID

Пример:

```
Запрос:
curl -X DELETE "<IP_HOST>:9010/api/v1/words/<Word_ID>" -H "accept: application/json"
```

- IP_HOST - ip хоста;
- Word_ID - id слова которое нужно удалить.

Рекомпиляция ASR-модели

Рекомпиляция модели — это процесс создания ASR-модели на основе уже существующей, в которую будут добавлены пользовательские слова и словосочетания.

Новые слова берутся из двух источников — телефонной книги и таблицы [extended_words](#), которой управляет ecss-clerk (автосекретарь). На основе телефонной книги строится корпус слов с перестановками, добавлением вариантов имен с сокращениями (которые берутся из таблицы [aliases](#)). То есть после рекомпиляции получается модель, в которой есть все имена из телефонной книги.

Если нужно добавить свои алиасы и дополнительные слова, то нужно воспользоваться [API добавление слов в базу](#).

После этого можно запустить recompilation, отправив на API соответствующий запрос.

Пример:

```
Запрос: curl localhost:9010/model/recompile

Ответ:
{
  "msg": "Recompile process start sucessfully"
}
```

Мониторинг за состоянием recompilation, осуществляется командой:

```
journalctl -u kaldi-ru.service
```

Также можно следить за recompilation через логи `/var/log/kaldi/model/build_model.log`

Процесс recompilation занимает 10-15 минут.

Если после recompilation, распознавание слов не улучшилось, то можно воспользоваться ручными настройками через таблицу [hard_aliases](#).