Руководство по инсталляции и начальному конфигурированию

ECSS-10, версия: 3.18

Содержание

1 Аннотация ... 12
2 Целевая аудитория ... 13
3 Разработка проекта системы ... 14
4 Подготовка сервера для инсталляции системы ... 17
5 Требования к серверам ... 18
6 Требования к виртуальным машинам ... 20
7 Подразделы ... 21
8 Первоначальная установка системы 3.18 с резервированием в кластере из двух серверов ... 22
8.1 Исходные данные ... 23
8.2 Включение в сеть ... 27
8.3 Последовательность установки SSW ECSS10 в кластере из двух серверов ... 28
8.3.1 Подготовительный этап ... 28
8.4 Установка ОС ... 29
8.5 Настройка сети ... 61
8.5.1 Пример netplan для режима active-backup ... 63
8.5.2 Пример netplan для режима 802.3ad ... 65
8.5.3 Настройка /etc/hosts ... 67
8.6 "Оптимизация" Операционной Системы ... 68
8.6.1 Выставить параметры ОС в режим производительности ... 68
8.6.2 Отключить SWAP ... 69
8.6.3 Установка часового пояса ... 70
8.6.4 Улучшение работы высоконагруженных серверов ... 70
8.7 Обновление программного обеспечения операционной системы ... 73
8.8 Установка и настройка программного обеспечения ... 75
8.9 Инсталляция обязательных пакетов ECSS ... 78
8.9.1 Установка пакета ecss-dns-env ... 78
8.9.2 Установка Postgres ... 79
8.9.3 Установка пакета ecss-postgres-bdr-ssw ... 80
8.9.4 Проверить корректность настройки репликации ... 81
8.9.5 Установка ecss-node ... 86
8.9.6 Конфигурация NTP ... 94
8.9.7 Установка лицензии ... 100
8.9.8 Установка остальных пакетов ecss и их настройка ... 110
8.9.9 Настройка VRRP для SIP-адаптера ... 122
8.9.10 Дальнейшая настройка ... 125

8.9.11 Настройка безопасности. SSH....125
8.9.12 Добавление услуг....126
8.10 Первоначальная конфигурация системы с помощью web-конфигуратора....127
8.10.1 Исходные данные....127
8.10.2 Подготовка к работе....127
8.10.3 Создание учетной записи оператора....128
8.10.4 Добавление медиа-ресурсов....132
8.10.5 Создание IP-set (sip-транспорта) и присвоение его к домену....134
8.10.6 Создание домена....135
8.10.7 Создание абонентов....140
8.10.8 Создание и применение контекстов маршрутизации для домена....143
8.10.9 Создание транка....148
8.10.10 Создание IVR-скрипта....150
8.10.11 Завершение настройки маршрутизации....152
8.11 Первоначальная конфигурация системы через интерфейс командной строки (CLI)....152
8.11.1 Добавление медиа-ресурсов в интерфейсе командной строки (CLI)....152
8.11.2 Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI)....156
8.11.3 Создание и настройка группы SIP-IP-адресов (IP-set)....159
8.11.4 Создание Домена и подключение IP-set....161
9 Первоначальная установка системы 3.18 без резервирования с одним сервером....164
9.1 Исходные данные....165
9.2 Включение в сеть....168
9.3 Последовательность установки SSW ECSS10 на одном сервере....170
9.3.1 Подготовительный этап....170
9.4 Установка ОС....170
9.5 Настройка сети....203
9.5.1 Подготовка сетевых интерфейсов системы....204
9.6 "Оптимизация" Операционной Системы....206
9.6.1 Выставить параметры ОС в режим производительности....206
9.6.2 Отключить SWAP....207
9.6.3 Установка часового пояса....208
9.6.4 Улучшение работы высоконагруженных серверов....208
9.7 Обновление программного обеспечения операционной системы....211
9.8 Установка и настройка программного обеспечения....212
9.8.1 Настройка пакета ecss-dns-env....215
9.8.2 Установка Postgres....215

9.8.3 Установка пакета ecss-postgres-bdr-ssw....215
9.8.4 Установка ecss-node....216
9.8.5 Установка лицензии....220
9.8.6 Установка пакетов ecss-restf, ecss-media-server, ecss-web-conf....230
9.8.7 Настройка snpmd....237
9.8.8 Настройка безопасности. SSH....237
9.8.9 Добавление услуг....238
9.9 Первоначальная конфигурация системы с помощью web-конфигуратора....238
9.9.1 Исходные данные....238
9.9.2 Подготовка к работе....239
9.9.3 Создание учетной записи оператора....240
9.9.4 Добавление медиа-ресурсов....243
9.9.5 Создание IP-set (sip-транспорта) и присвоение его к домену....245
9.9.6 Создание домена....247
9.9.7 Создание абонентов....251
9.9.8 Создание и применение контекстов маршрутизации для домена....254
9.9.9 Создание транка....259
9.9.10 Создание IVR-скрипта....261
9.9.11 Завершение настройки маршрутизации....263
9.10 Первоначальная конфигурация системы через интерфейс командной строки (CLI)....263
9.10.1 Добавление медиа-ресурсов в интерфейсе командной строки (CLI)....263
9.10.2 Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI) CoCon....266
9.10.3 Создание и настройка группы SIP-IP-адресов (IP-set)....269
9.10.4 Создание Домена и подключение IP-set....270
10 Установка ECSS10 3.18 на Astra Linux, в открытом контуре, установка в кластере....273
10.1 Первоначальная установка системы с резервированием, в кластере из двух серверов....273
10.1.1 Исходные данные....273
10.1.2 Последовательность установки SSW ECSS10 в кластере из двух серверов....276
10.1.3 Установка ОС....277
10.1.4 Проверка установки ОС....284
10.1.5 Настройка /etc/hosts....285
10.2 Конфигурация репозиториев....285
10.2.1 Изменение приоритета репозиториев....286
10.2.2 "Оптимизация" Операционной Системы....287
10.3 Настройка сетевых интерфейсов....292

10.3.1 Пример netplan 293
10.4 Обновление ОС и инсталляция необходимого ПО 294
10.4.1 Обновление системы 294
10.4.2 Инсталляция необходимого ПО 295
10.5 Инсталляция пакетов ECSS 298
10.5.1 Предварительные требования 298
10.5.2 Инсталляция обязательных пакетов ECSS 300
10.6 Первоначальная конфигурация системы с помощью web-конфигуратора 344
10.6.1 Исходные данные 344
10.6.2 Подготовка к работе 345
10.6.3 Создание учетной записи оператора 346
10.6.4 Добавление медиа-ресурсов 349
10.6.5 Создание IP-set (sip-транспорта) и присвоение его к домену 351
10.6.6 Создание домена 353
10.6.7 Создание абонентов 358
10.6.8 Создание и применение контекстов маршрутизации для домена 361
10.6.9 Создание транка 366
10.6.10 Создание IVR-скрипта 368
10.6.11 Завершение настройки маршрутизации 370
10.7 Первоначальная конфигурация системы через интерфейс командной строки (CLI) 370
10.7.1 Добавление медиа-ресурсов в интерфейсе командной строки (CLI) 370
10.7.2 Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI) 373
10.7.3 Создание Домена и подключение IP-set 378
10.7.4 Добавление услуг 378
11 Установка ECSS10 3.18 на Astra Linux, в открытом контуре, на одиночный сервер 381
11.1 Первоначальная установка системы на одном сервере 381
11.1.1 Исходные данные 381
11.1.2 Последовательность установки SSW ECSS10 без резервирования с одним сервером 383
11.1.3 Установка ОС 384
11.1.4 Проверка установки ОС 391
11.1.5 Настройка /etc/hosts 391
11.2 Конфигурация репозиториев 392
11.2.1 Изменение приоритета репозиториев 392
11.2.2 "Оптимизация" Операционной Системы 393
11.3 Настройка сетевых интерфейсов 398

11.3.1 Пример netplan 398
11.4 Обновление ОС и инсталляция необходимого ПО 399
11.4.1 Обновление системы 399
11.4.2 Инсталляция необходимого ПО 399
11.5 Инсталляция пакетов ECSS 401
11.5.1 Предварительные требования 401
11.5.2 Инсталляция обязательных пакетов 403
11.6 Первоначальная конфигурация системы с помощью web-конфигуратора 427
11.6.1 Исходные данные 427
11.6.2 Подготовка к работе 427
11.6.3 Создание учетной записи оператора 428
11.6.4 Добавление медиа-ресурсов 432
11.6.5 Создание IP-set (sip-транспорта) и присвоение его к домену 434
11.6.6 Создание домена 435
11.6.7 Создание абонентов 440
11.6.8 Создание и применение контекстов маршрутизации для домена 443
11.6.9 Создание транка 448
11.6.10 Создание IVR-скрипта 450
11.6.11 Завершение настройки маршрутизации 452
11.7 Первоначальная конфигурация системы через интерфейс командной строки (CLI) 452
11.7.1 Добавление медиа-ресурсов в интерфейсе командной строки (CLI) 452
11.7.2 Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI) 454
11.7.3 Создание Домена и подключение IP-set 457
12 Установка ECSS10 3.18 на Astra Linux, в закрытом контуре, установка в кластере 460
12.1 Первоначальная установка системы с резервированием, в кластере из двух серверов 460
12.1.1 Исходные данные 460
12.1.2 Последовательность установки SSW ECSS10 в кластере из двух серверов 464
12.1.3 Установка ОС 464
12.1.4 Проверка установки ОС 471
12.1.5 Настройка /etc/hosts 472
12.2 Конфигурация репозиториев 472
12.2.1 Изменение приоритета репозиториев 473
12.2.2 "Оптимизация" Операционной Системы 474
12.3 Настройка сетевых интерфейсов 479
12.3.1 Пример netplan для режима active-backup 480

12.4 Обновление ОС и инсталляция необходимого ПО 481
12.4.1 Обновление системы 481
12.4.2 Инсталляция необходимого ПО 482
12.5 Инсталляция пакетов ECSS 485
12.5.1 Предварительные требования 485
12.5.2 Инсталляция обязательных пакетов 487
12.6 Первоначальная конфигурация системы с помощью web-конфигуратора 530
12.6.1 Исходные данные 530
12.6.2 Подготовка к работе 531
12.6.3 Создание учетной записи оператора 532
12.6.4 Добавление медиа-ресурсов 535
12.6.5 Создание IP-set (sip-транспорта) и присвоение его к домену 537
12.6.6 Создание домена 539
12.6.7 Создание абонентов 544
12.6.8 Создание и применение контекстов маршрутизации для домена 547
12.6.9 Создание транка 552
12.6.10 Создание IVR-скрипта 554
12.6.11 Завершение настройки маршрутизации 556
12.7 Первоначальная конфигурация системы через интерфейс командной строки (CLI) 556
12.7.1 Добавление медиа-ресурсов в интерфейсе командной строки (CLI) 556
12.7.2 Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI) 559
12.7.3 Создание Домена и подключение IP-set 564
12.7.4 Добавление услуг 564
13 Установка ECSS10 3.18 на Astra Linux, в закрытом контуре, на одиночный сервер 567
13.1 Первоначальная установка системы на одном сервере 567
13.1.1 Исходные данные 567
13.1.2 Последовательность установки SSW ECSS10 без резервирования с одним сервером 569
13.1.3 Установка ОС 570
13.1.4 Проверка установки ОС 577
13.1.5 Настройка /etc/hosts 577
13.2 Конфигурация репозиториев 578
13.2.1 Изменение приоритета репозиториев 579
13.2.2 "Оптимизация" Операционной Системы 579
13.3 Настройка сетевых интерфейсов 584
13.3.1 Пример netplan 584

13.4 Инсталляция необходимого ПО 585
13.4.1 Установка дополнительных пакетов программ 585
13.5 Инсталляция пакетов ECSS 587
13.5.1 Предварительные требования 587
13.5.2 Инсталляция обязательных пакетов 589
13.6 Первоначальная конфигурация системы с помощью web-конфигуратора 612
13.6.1 Исходные данные 612
13.6.2 Подготовка к работе 612
13.6.3 Создание учетной записи оператора 613
13.6.4 Добавление медиа-ресурсов 617
13.6.5 Создание IP-set (sip-транспорта) и присвоение его к домену 619
13.6.6 Создание домена 620
13.6.7 Создание абонентов 625
13.6.8 Создание и применение контекстов маршрутизации для домена 628
13.6.9 Создание транка 633
13.6.10 Создание IVR-скрипта 635
13.6.11 Завершение настройки маршрутизации 637
13.7 Первоначальная конфигурация системы через интерфейс командной строки (CLI) 637
13.7.1 Добавление медиа-ресурсов в интерфейсе командной строки (CLI) 637
13.7.2 Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI) 639
13.7.3 Создание Домена и подключение IP-set 642
14 Пример первичной конфигурации системы с помощью web-конфигуратора 645
14.1 Исходные данные 645
14.2 Подготовка к работе 645
14.3 Создание учетной записи оператора 646
14.4 Создание домена 650
14.5 Создание IP-set (sip-транспорта) и присвоение его к домену 654
14.6 Создание абонентов 657
14.7 Создание и применение контекстов маршрутизации для домена 660
14.8 Создание транка 663
14.9 Создание IVR-скрипта 665
14.10 Завершение настройки маршрутизации 667
14.11 Настройка услуг 668
14.12 Добавление медиа-ресурсов 669
15 Чек-лист по установке ECSS-10 670
16 Регламент обновления/деинсталляции ECSS 682

16.1 Обновление пакета ecss-postgres-bdr-ssw 682
16.2 Обновление с версии 3.14.14 на 3.14.15 685
16.2.1 Обновление с версии 3.14.14 на 3.14.15 Подготовка к обновлению системы Установка ECSS-10 версии 3.14.15 Установка ecss-mysql Установка ecss-node Установка остальных пакетов Обновление второго сервера, если система в кластере Установка ecss-mysql Установка ecss-node Установка остальных пакетов Сборка кластера Проверка работы системы в кластере Деинсталляция компонентов ECSS Удаление ecss-mysql Удаление пакета ecss-node Удаление пакета ecss-media-server Удаление ненужных пакетов Удаление пакетов по маске Команды полной деинсталляции ECSS Debconf 685
16.2.2 Обновление с версии 3.14.14 на 3.14.15 685
16.2.3 Обновление второго сервера, если система в кластере 694
16.2.4 Деинсталляция компонентов ECSS 699
16.3 Обновление с версии 3.14.15 на 3.14.16 Astra 701
16.3.1 Обновление с версии 3.14.15 на 3.14.16 702
16.4 Обновление с версии 3.14.15 на 3.14.16 Ubuntu 728
16.4.1 Обновление с версии 3.14.15 на 3.14.16 729
16.5 Обновление с версии 3.14.16 на 3.17 775
16.5.1 Обновление с версии 3.14.16 на 3.17 775
16.6 Обновление с версии 3.17.1 на 3.18 804
16.6.1 1 Подготовка хостов к обновлению 804
16.6.2 2 Подготовка к обновлению сервисов SSW 807
16.6.3 3 Обновление сервисов SSW на ecss2 807
16.6.4 4 Подготовительные действия на ecss1 819
16.6.5 5 Обновление сервисов SSW на ecss1 820
16.6.6 7 Проверка обновления 832
17 Начальная настройка конфигурации 833
17.1 Понятия и определения 833
17.2 Предварительные условия 834
17.3 Настройка ECSS-10 для производительных систем 834
17.3.1 Выделение отдельных ядер процессора для MSR 835
17.3.2 Установка работы CPU в режим perfomance 836
17.3.3 Запуск MSR на изолированных ядрах процессора 837
17.3.4 Настройка использования определенных ядер процессора для erlang-based служб 837
17.4 Порядок начальной настройки конфигурации 839
17.4.1 Настройка программного медиасервера 840
17.4.2 Создание и настройка группы SIP-IP-адресов (IP-set) 840
17.4.3 Создание и настройка доменов (виртуальных АТС) 841
17.4.4 Настройка дополнительных сервисов 848
18 Рекомендации по обеспечению безопасности системы 850

18.1 Общие требования 850
18.1.1 Telnet 851
18.1.2 Отключить доступ через root на сервер 851
18.1.3 Включить доступ только по ключам ssh 851
18.1.4 Использовать безопасные ключи rsa -b 4096 851
18.1.5 Автоматическая генерация паролей для SIP-пользователей 851
18.1.6 Изменение паролей пользователей 852
18.1.7 Использовать SBC, настроить firewall, ACL 865
18.1.8 Обновлять Ubuntu и утилиты сервера при выходе пакета безопасности 865
18.1.9 NTP 865
18.2 Настройка безопасности OpenSSH 865
18.2.1 Настройка конфигурации сервера и клиента ssh 865
19 Интеграция SSW и ELM сервером 872
19.1 Типы лицензирования 872
19.2 Установка лицензирования 872
19.2.1 Установка default лицензирования 872
19.2.2 Установка distributed лицензирования 873
20 Установка Postgres BDR v.3.17 881
20.1 Установка Установка вне кластера Установка пакета ecss-postgres-bdr-ssw Проверка Установка в кластере Установка пакета ecss-postgres-bdr-ssw Проверить корректность настройки репликации Удаление Postgres BDR Восстановления кластера 881
20.2 Установка 881
20.2.1 Установка вне кластера 882
20.2.2 Установка в кластере 883
20.3 Удаление Postgres BDR 889
20.3.1 Восстановления кластера 890

- Аннотация
- Целевая аудитория
- Разработка проекта системы
- Подготовка сервера для инсталляции системы
- Требования к серверам
- Требования к виртуальным машинам
- Подразделы

1 Аннотация

В данной части руководства приводится информация о разработке проекта развертывания, установке и первоначальной настройке программно-аппаратного комплекса ECSS-10 (в дальнейшем именуемого ECSS-10, комплекс или система).

2 Целевая аудитория

Руководство предназначено для технического персонала, выполняющего первичную установку и настройку ECSS-10. Квалификация технического персонала предполагает знание основ работы стеков протоколов ОКС-7, TCP/IP, UDP/IP, SIP, SIP-T, SIGTRAN, MEGACO, принципов построения Ethernet-сетей и практические навыки работы с UNIX-подобными ОС (Debian, Ubuntu, freeBSD, Red Hat Linux).

3 Разработка проекта системы

В данной главе приводится информация для начала разработки проект-схемы будущей системы.

Проект схемы выполняется перед инсталляцией системы.

При разработке проект-схемы необходимо знать:

- функциональный состав системы: протоколы, по которым будет осуществляться подключение шлюзов к ECSS-10, способы получения тарификационной и статистической информации;
- набор услуг, предоставляемый системой, помимо коммутации телефонных вызовов, уровень расчетной нагрузки по вызовам с услугами;
- предполагаемый уровень расчетной нагрузки на систему: количество вызовов в секунду на определенных направлениях, суммарная нагрузка на систему;
- необходимый уровень резервирования системы;
- требования по обеспечению поддержки СОРМ.

В зависимости от функциональных требований системы составляется перечень необходимых подсистем ECSS.

Обязательными компонентами ECSS являются:

- кластер Storage;
- кластер BUS;
- кластер Core;
- кластер Mediator;
- кластер Adapter SIP.

В зависимости от функциональных требований выбираются следующие компоненты:

- кластер Adapter Megaco,
- кластер Adapter Sigtran.

Исходя из уровня расчетной нагрузки на систему и требований по резервированию выбираются:

- количество и характеристики аппаратных ресурсов (тип процессора, количество RAM, размер дискового массива, количество сетевых интерфейсов), на которых будет разворачиваться система;
- топология распределения кластеров на аппаратных ресурсах (серверах) системы.

Далее при известной информации об аппаратных ресурсах и распределении кластеров на этих ресурсах производится проектирование сетевого уровня системы. Интерфейсам назначаются сетевые адреса, определяется топология связи серверов между собой, исходя из требований по резервированию.

В итоговом документе, описывающем систему, должна содержаться информация, приведенная в таблицах 1–3.

Таблица 1. Аппаратная конфигурация системы

Компонент	**Модель**	**Характеристики**
сервер 1	HP G360	CPU, RAM, HDD, Ethernet count
сервер 2	HP G360	CPU, RAM, HDD, Ethernet count
...	...	...
сервер n	HP G360	CPU, RAM, HDD, Ethernet count

Таблица 2. Топология сетевых соединений

Сервер	Имя сервера	Интерфейс	IP	MASK	Gateway	connected to
сервер 1	ecss1	Eth 0	IP	MASK	Gateway	Switch 1:port 1 или сервер 2:Ethx
сервер 1	ecss1	Eth 1	IP	MASK	Gateway	Switch 1:port 2 или сервер 2:Eth1
сервер 2	ecss2	Eth 0	IP	MASK	Gateway	Switch 1:port 3 или сервер 1:Ethn
сервер 2	ecss2	Eth 1	IP	MASK	Gateway	Switch 1:port 4 или сервер 1:Eth1

Таблица 3. Топология программных компонентов

Сервер	Storage	BUS	Core	Mediator	Adapter SIP	Adapter Megaco	Adapter Sigtran
сервер 1	ds1@ecss1	mycelium1@ecss1	core1@ecss1	md1@ecss1	sip1@ecss1		
сервер 2	ds1@ecss2	mycelium1@ecss2	core1@ecss2	md1@ecss2	sip1@ecss2		

Удобным средством визуализации представленной выше информации является структурная схема ECSS.

Таблица 4. Аппаратная конфигурация системы

Компонент	Модель	Характеристики
сервер 1	HP Proliant DL360 G7	E5620 QuadCore, 2x2Gb, 2x300Gb 4xGbEth
сервер 2	HP Proliant DL360 G7	E5620 QuadCore, 2x2Gb, 2x300Gb 4xGbEth

Таблица 5. Топология сетевых соединений

Сервер	Имя сервера	Интерфейс	IP	MASK	Gateway	connected to
сервер 1	ecss1	Eth 0				Switch 1:port 1
сервер 1	ecss1	Eth 1				Switch 2:port 1
сервер 1	ecss1	Eth 2				Switch 1:port 2

Сервер	Имя сервера	Интерфейс	IP	MASK	Gateway	connected to
сервер 1	ecss1	Eth 3				Switch 2:port 2
сервер 2	ecss2	Eth 0				Switch 1:port 3
сервер 2	ecss2	Eth 1				Switch 2:port 3
сервер 2	ecss2	Eth 2				Switch 1:port 4
сервер 2	ecss2	Eth 3				Switch 2:port 4

4 Подготовка сервера для инсталляции системы

Предварительно необходимо согласовать сетевые интерфейсы, разработать схему сети, подготовить план нумерации.

На данном этапе осуществляется подготовка серверов к инсталляции программных компонентов ECSS-10. Подготовка серверов заключается в установке операционной системы с заданным набором пакетов и их последующей настройке.

Программные компоненты системы ECSS-10 являются кроссплатформенными. Потенциально поддерживаются различные операционные системы, но на данный момент официально протестированной и поддерживаемой ОС является Linux Ubuntu Server 22.04 LTS/ Astra Linux 1.8, порядок установки и настройки ОС описаны в Приложении .

5 Требования к серверам

Актуальная информация с требованиями к серверам, которые должны использоваться для работы комплекса ECSS-10.

№	Требования к серверам SSW	Серия Light+	Серия Midi	Серия Heavy	Серия Супер Heavy	Серия Top (1)
1	**Характеристики системы**					
1.1	Максимальное количество абонентов	5000	10000	20000	40000	-
1.2	Максимальная нагрузка одновременных соединений class 5	800	1500	3000	6000	-
1.3	Максимальная нагрузка одновременных соединений class 4	2400	4500	9000	20000	80000
2	**Характеристики сервера**					
2.1	Марка	HP (Lenovo)	HP (Lenovo)	HP (Lenovo)	HP (Lenovo)	HP /Lenovo / Depo
2.2	Серия	DL20 Gen10/DL 360 Gen10 (SR530)	DL360 Gen10 (SR530/SR630)	DL360 Gen10 (SR630)	DL360 Gen10 (SR630)	DL360 Gen10 / SR650 V2 / Storm 3450
2.3	Процессор	Intel Xeon 4214	Intel Xeon 5220	Intel Xeon 6240	Intel Xeon 8268	Intel Xeon 8380
2.4	Количество процессоров	1	1	2	2	2
2.5	Оперативная память	16 Gb	24 Gb	32 Gb	64 Gb	1024 Gb
2.6	HDD	От 3X500 SATA (от 7200 rpm)	От 3x300 Gb SAS (от 10000 rpm)	От 3x600 Gb SAS (от 10000 rpm) 2x150Gb SSD	От 6x800 Gb SSD, 2x300Gb M.2 SSD	От 6x1.2 Tb SSD, 2x300Gb M.2 SSD
2.7	RAID	no raid board	HW Raid, от 1Gb cache+battery	HW Raid, от 1Gb cache+battery	HW Raid, от 2Gb Flash cache, поддержка RAID-5	HW Raid, от 2Gb Flash cache, поддержка RAID-5

№	Требования к серверам SSW	Серия Light+	Серия Midi	Серия Heavy	Серия Супер Heavy	Серия Top (1)
3	**Дополнительные комплектующие сервера (в базовый комплект не входят)**					
3.1	Лицензия на удалённое управление	по желанию	+	+	+	+
3.2	Резервный источник питания	по желанию	+	+	+	+
3.3	Организация хранилища записей разговоров	Дополнительные HDD, объединенные в RAID-5	Лицензия на HW Raid с поддержкой RAID-5, дополнительные HDD для хранения записей	Лицензия на HW Raid с поддержкой RAID-5, дополнительные HDD для хранения записей	Лицензия на HW Raid с поддержкой RAID-5, дополнительные HDD для хранения записей	Лицензия на HW Raid с поддержкой RAID-5, дополнительные HDD для хранения записей

Полоса пропускания СПД

Требуемая полоса пропускания СПД **не менее** 1000Mb/s

6 Требования к виртуальным машинам

Минимальные требования при установке ECSS-10 на виртуальные машины*

№	Требования к серверам SSW	Уровень Light (2)(3)	Уровень Light+	Уровень Midi	Уровень Heavy	Уровень Супер Heavy	Уровень Top (1)
1	**Характеристики системы**						
1.1	Максимальное количество абонентов	1000	5000	10000	20000	40000	-
1.2	Максимальная нагрузка одновременных соединений class 5	250	800	1500	3000	6000	-
1.3	Максимальная нагрузка одновременных соединений class 4	1000	4500	9000	20000	80000	-
2	**Характеристики сервера**						
2.4	Количество ядер/потоков (vCore)	**8C**	**12C/24T**	**18C/36T**	**36C/72T**	**48C/96T**	**80C/160T** и более
2.5	Частота процессора	от 2.4 Ghz	от 3 Ghz	от 3Ghz	от 3.6 Ghz	от 3.4 Ghz	от 3.2 Ghz
2.6	Оперативная память	8 Gb	16 Gb	24 Gb	32 Gb	от 64 Gb	от 512 Gb
2.7	HDD	От 50 GB, 75 IOPS	От 500 GB SATA (от 7200 rpm, от 75 IOPS)	От 500 Gb SAS (от 10000 rpm, 170 IOPS)	От 500 Gb SAS (от 10000 rpm, 170 IOPS) и 150Gb SSD	от 800 Gb SSD и 300Gb M.2 SSD	От 1.2 Tb SSD и 300Gb M.2 SSD

C - core, они же ядра

T - thread = поток, vCore

(1) — серия Top на текущий момент рассчитана для реализации контроллера транковых шлюзов (работа по 4 классу).
(2) — уровень Light применяется только для расчета виртуальных машин, для физических серверов минимальная рекомендуемая конфигурация — Light+.
(3) — уровень Light является минимальным рекомендуемым. При запросе расчета ВМ для малого числа абонентов (до 1000: 100, 200...) используем его.

*Если на системе планируется использование таких сервисов как КЦ, запись разговоров, видео-вызовы, то необходимо согласовать требования с технической поддержкой

7 Подразделы

- Первоначальная установка системы 3.18 с резервированием в кластере из двух серверов
- Первоначальная установка системы 3.18 без резервирования с одним сервером
- Установка ECSS10 3.18 на Astra Linux, в открытом контуре, установка в кластере
- Установка ECSS10 3.18 на Astra Linux, в открытом контуре, на одиночный сервер
- Установка ECSS10 3.18 на Astra Linux, в закрытом контуре, установка в кластере
- Установка ECSS10 3.18 на Astra Linux, в закрытом контуре, на одиночный сервер
- Пример первичной конфигурации системы с помощью web-конфигуратора
- Чек-лист по установке ECSS-10
- Регламент обновления/деинсталляции ECSS
- Начальная настройка конфигурации
- Рекомендации по обеспечению безопасности системы
- Интеграция SSW и ELM сервером
- Установка Postgres BDR v.3.17

8 Первоначальная установка системы 3.18 с резервированием в кластере из двух серверов

- Исходные данные
- Включение в сеть
- Последовательность установки SSW ECSS10 в кластере из двух серверов
  - Подготовительный этап
    - Предварительные требования
- Установка ОС
- Настройка сети
  - Пример netplan для режима active-backup
  - Пример netplan для режима 802.3ad
  - Настройка /etc/hosts
- "Оптимизация" Операционной Системы
  - Выставить параметры ОС в режим производительности
  - Отключить SWAP
  - Установка часового пояса
  - Улучшение работы высоконагруженных серверов
- Обновление программного обеспечения операционной системы
- Установка и настройка программного обеспечения
- Инсталляция обязательных пакетов ECSS
  - Установка пакета ecss-dns-env
  - Установка Postgres
  - Установка пакета ecss-postgres-bdr-ssw
  - Проверить корректность настройки репликации
  - Установка ecss-node
    - Установка на сервере ecss1
    - Установка на сервере ecss2
  - Конфигурация NTP
    - Синхронизация времени на серверах
    - Установка и настройка NTP
  - Установка лицензии
    - Конфигурация LM
    - Подключение SSW к Менеджера Лицензий
  - Установка остальных пакетов ecss и их настройка
    - glusterfs
    - Установка пакета ecss-restfs и его настройка
    - ecss-media-server
    - ecss-web-conf
  - Настройка VRRP для SIP-адаптера
  - Дальнейшая настройка
    - snpmd
  - Настройка безопасности. SSH
  - Добавление услуг
- Первоначальная конфигурация системы с помощью web-конфигуратора
  - Исходные данные
  - Подготовка к работе
  - Создание учетной записи оператора
  - Добавление медиа-ресурсов.
  - Создание IP-set (sip-транспорта) и присвоение его к домену
  - Создание домена
  - Создание абонентов
  - Создание и применение контекстов маршрутизации для домена
  - Создание транка

    - Создание IVR-скрипта
    - Завершение настройки маршрутизации
- Первоначальная конфигурация системы через интерфейс командной строки (CLI)
    - Добавление медиа-ресурсов в интерфейсе командной строки (CLI)
    - Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI)
    - Создание и настройка группы SIP-IP-адресов (IP-set)
    - Создание Домена и подключение IP-set

8.1 Исходные данные

Исходные данные

В данном разделе использованы специфичные для данного продукта термины, ознакомиться с ними можно здесь.
Техническое задание:
Требуется интеграция программного коммутатора 5 класса ECSS-10 (SSW) на 2-х физических серверах с поддержкой SIP со следующими параметрами на нагрузку группы серверов:

- Максимальное количество абонентов 15000 (MUL — Max user limit);
- Максимальное количество одновременных соединений 2000 (MCL — Max call limit);
- Требуется полный резерв системы ecss (все ноды резервируется на втором сервере);
- Количество сетевых интерфейсов ethernet — 4.

По техническому заданию требуется определить аппаратную платформу.

Актуальная информация с требованиями к серверам, которые должны использоваться для работы комплекса ECSS-10.

№	Требования к серверам SSW	Серия Light+	Серия Midi	Серия Heavy	Серия Супер Heavy	Серия Top (1)
1	**Характеристики системы**					
1.1	Максимальное количество абонентов	5000	10000	20000	40000	-
1.2	Максимальная нагрузка одновременных соединений class 5	800	1500	3000	6000	-
1.3	Максимальная нагрузка одновременных соединений class 4	2400	4500	9000	20000	80000
2	**Характеристики сервера**					

№	Требования к серверам SSW	Серия Light+	Серия Midi	Серия Heavy	Серия Супер Heavy	Серия Top (1)
2.1	Марка	HP (Lenovo)	HP (Lenovo)	HP (Lenovo)	HP (Lenovo)	HP /Lenovo / Depo
2.2	Серия	DL20 Gen10/DL 360 Gen10 (SR530)	DL360 Gen10 (SR530/SR630)	DL360 Gen10 (SR630)	DL360 Gen10 (SR630)	DL360 Gen10 / SR650 V2 / Storm 3450
2.3	Процессор	Intel Xeon 4214	Intel Xeon 5220	Intel Xeon 6240	Intel Xeon 8268	Intel Xeon 8380
2.4	Количество процессоров	1	1	2	2	2
2.5	Оперативная память	16 Gb	24 Gb	32 Gb	64 Gb	1024 Gb
2.6	HDD	От 3X500 SATA (от 7200 rpm)	От 3x300 Gb SAS (от 10000 rpm)	От 3x600 Gb SAS (от 10000 rpm) 2x150Gb SSD	От 6x800 Gb SSD, 2x300Gb M.2 SSD	От 6x1.2 Tb SSD, 2x300Gb M.2 SSD
2.7	RAID	no raid board	HW Raid, от 1Gb cache+battery	HW Raid, от 1Gb cache+battery	HW Raid, от 2Gb Flash cache, поддержка RAID-5	HW Raid, от 2Gb Flash cache, поддержка RAID-5
3	**Дополнительные комплектующие сервера (в базовый комплект не входят)**					
3.1	Лицензия на удалённое управление	по желанию	+	+	+	+
3.2	Резервный источник питания	по желанию	+	+	+	+
3.3	Организация хранилища записей разговоров	Дополнительные HDD, объединенные в RAID-5	Лицензия на HW Raid с поддержкой RAID-5, дополнительные HDD для хранения записей	Лицензия на HW Raid с поддержкой RAID-5, дополнительные HDD для хранения записей	Лицензия на HW Raid с поддержкой RAID-5, дополнительные HDD для хранения записей	Лицензия на HW Raid с поддержкой RAID-5, дополнительные HDD для хранения записей

⚠ **Полоса пропускания СПД**

Требуемая полоса пропускания СПД **не менее** 1000Mb/s

Минимальные требования при установке ECSS-10 на виртуальные машины*

№	Требования к серверам SSW	Уровень Light (2)(3)	Уровень Light+	Уровень Midi	Уровень Heavy	Уровень Супер Heavy	Уровень Top (1)
1	**Характеристики системы**						
1.1	Максимальное количество абонентов	1000	5000	10000	20000	40000	-
1.2	Максимальная нагрузка одновременных соединений class 5	250	800	1500	3000	6000	-
1.3	Максимальная нагрузка одновременных соединений class 4	1000	4500	9000	20000	80000	-
2	**Характеристики сервера**						
2.4	Количество ядер/потоков (vCore)	**8C**	**12C/24T**	**18C/36T**	**36C/72T**	**48C/96T**	**80C/160T** и более
2.5	Частота процессора	от 2.4 Ghz	от 3 Ghz	от 3Ghz	от 3.6 Ghz	от 3.4 Ghz	от 3.2 Ghz
2.6	Оперативная память	8 Gb	16 Gb	24 Gb	32 Gb	от 64 Gb	от 512 Gb
2.7	HDD	От 50 GB, 75 IOPS	От 500 GB SATA (от 7200 rpm, от 75 IOPS)	От 500 Gb SAS (от 10000 rpm, 170 IOPS)	От 500 Gb SAS (от 10000 rpm, 170 IOPS) и 150Gb SSD	от 800 Gb SSD и 300Gb M.2 SSD	От 1.2 Tb SSD и 300Gb M.2 SSD

C - core, они же ядра

T - thread = поток, vCore

(1) — серия Top на текущий момент рассчитана для реализации контроллера транковых шлюзов (работа по 4 классу).
(2) — уровень Light применяется только для расчета виртуальных машин, для физических серверов минимальная рекомендуемая конфигурация — Light+.
(3) — уровень Light является минимальным рекомендуемым. При запросе расчета ВМ для малого числа абонентов (до 1000: 100, 200...) используем его.

*Если на системе планируется использование таких сервисов как КЦ, запись разговоров, видео-вызовы, то необходимо согласовать требования с технической поддержкой

Пример составления аппаратных требований:

Устройство	Требуемый ресурс		Серия аппаратного продукта
	MCL	MUL	
Сервер 1	2000	15000	Heavy
Сервер 2			Heavy

После определения по требованиям проекта составляем предварительную сетевую карту.

Пример разнесения составляющих по адресному пространству для одной ноды:

Имя сервера (хоста)	Роль	Интерфейс	Адрес	Адрес вашего проекта	Порт
Статические адреса программного коммутатора					
ecss1	Интерфейс управления сервером (порт 2000 ssh)	net.10	10.0.10.51/24		2000
ecss2	Интерфейс управления сервером (порт 2000 ssh)	net.10	10.0.10.52/24		2000
ecss1	Адрес ядра (ecss-core)	net.20	10.0.20.51/24		5000
ecss2	Адрес ядра (ecss-core)	net.20	10.0.20.52/24		5000
ecss1, ecss2	Адрес шлюза	net.10	10.0.10.1		-
ecss1, ecss2	Адреса DNS-серверов	net.10	10.0.10.1, 8.8.8.8		-
ecss1, ecss2	Адреса NTP-серверов	net.10	10.136.16.211, 10.136.16.212		123
Виртуальные адреса программного коммутатора					
ecss1	Виртуальный адрес программного адаптера хоста ecss1	net.20:SIP1	10.0.20.31/24		-

ecss2	Виртуальный адрес программного адаптера хоста ecss2	net.20:SIP2	10.0.20.32/24		-
ecss1	Запасной виртуальный адрес программного адаптера хоста ecss2 на хосте ecss1	net.20:SIP2	10.0.20.32/24		-
ecss2	Запасной виртуальный адрес программного адаптера хоста ecss1 на хосте ecss2	net.20:SIP1	10.0.20.31/24		-

8.2 Включение в сеть

Топологию включения сервера в сеть для обеспечения резервирования рекомендуется делать с использованием 2-х коммутаторов.

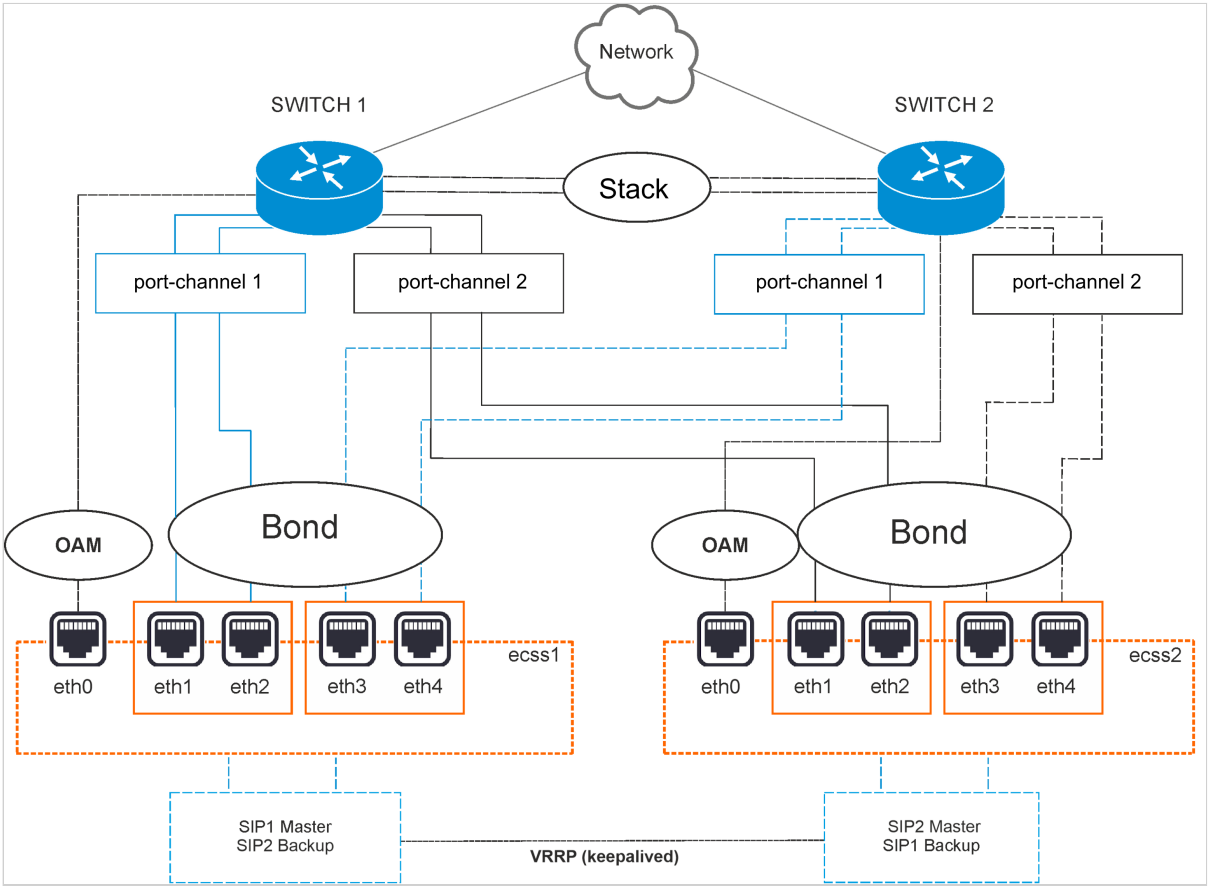


Рисунок 2 — Схема организации связи

Вариант 1. Схема active-backup

Коммутаторы связаны между собой в erps-кольцо.

Все 4 физических сетевых интерфейса каждого сервера объединяются в 1 агрегированный линк (bond). Агрегация портов сервера настраивается в режиме **active-backup**, т.е. в работе всегда находится только 1 сетевой интерфейс. Сетевые интерфейсы каждого сервера попарно включены в коммутаторы, на которых агрегация портов (port-channel) также настраивается в режиме active-backup. Например, eth0 и eth1 каждого сервера включаются в первый коммутатор (port-channel 1 и 2), а eth2 и eth3 (port-channel 3 и 4) — во второй.

Вариант 2. Схема LACP

Коммутаторы связаны между собой в стек. Стек должен логически работать как один коммутатор, способный обеспечивать агрегацию портов в режиме LACP между разными физическими коммутаторами. Пример — коммутаторы MES-3124 со специализированной прошивкой.

Все 4 физических сетевых интерфейса также объединяются в 1 агрегированный линк (bond). Агрегация портов сервера настраивается в режиме **802.3ad**. Создаются агрегированные группы сетевых карт с одинаковой скоростью и дуплексом. При таком объединении передача задействует все каналы в активной агрегации согласно стандарту IEEE 802.3ad. Выбор того, через какой интерфейс необходимо отправлять пакет, определяется политикой. По умолчанию это XOR-политика, можно использовать «xmit_hash» политику. Подробнее — в разделе Приложение E. Netplan.

Требования:

- Поддержка Ethtool в драйвере для получения информации о скорости и дуплексе на каждом сетевом интерфейсе;
- Поддержка на коммутаторе стандарта IEEE 802.3ad;

Сетевые интерфейсы сервера также попарно включены в коммутаторы, на которых агрегация портов (port-channel) тоже настраивается в режиме LACP. Например, eth0 и eth1 каждого сервера включаются в первый коммутатор (port-channel 1 и 2), а eth2 и eth3 (port-channel 3 и 4) — во второй.

8.3 Последовательность установки SSW ECSS10 в кластере из двух серверов

Установка SSW ECSS10 состоит из 2-х основных частей :

- Подготовительной - Установка ОС Ubuntu 22.04, обновлений пакетов ОС Ubuntu , "оптимизация" ОС, установка дополнительных пакетов программ, подготовка сетевых интерфейсов, /etc/hosts, ssh-keygen
- Установка приложений Eltex ECSS10

8.3.1 Подготовительный этап

В этом разделе приведено описание инсталляции операционной системы, а также необходимых и дополнительных пакетов. Система ECSS-10 версии 3.18 работает под управлением ОС Ubuntu 22.04.

Предварительные требования

- Установочный загрузочный носитель с дистрибутивом операционной системы;
- Подготовленный сервер с обновленным BIOS, ILO (если есть), подключенная сеть для доступа в сеть Интернет;

- Выставленный первый приоритет загрузки с установочного носителя — USB Flash или CD/DVD в BIOS;
- Достаточный объем дискового пространства и памяти в соответствии с проектом.

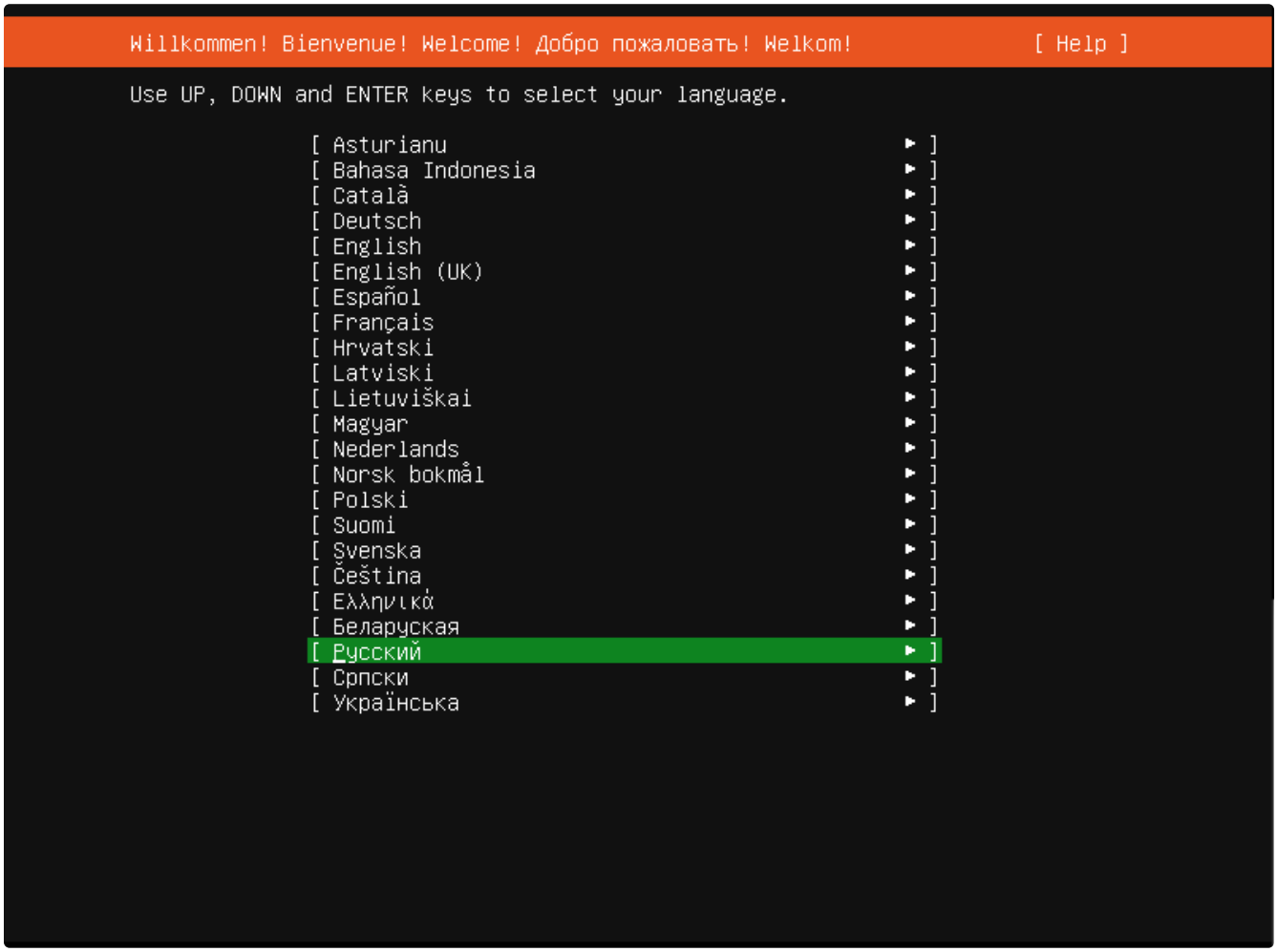
8.4 Установка ОС

Для установки ОС необходимо выполнить следующее:

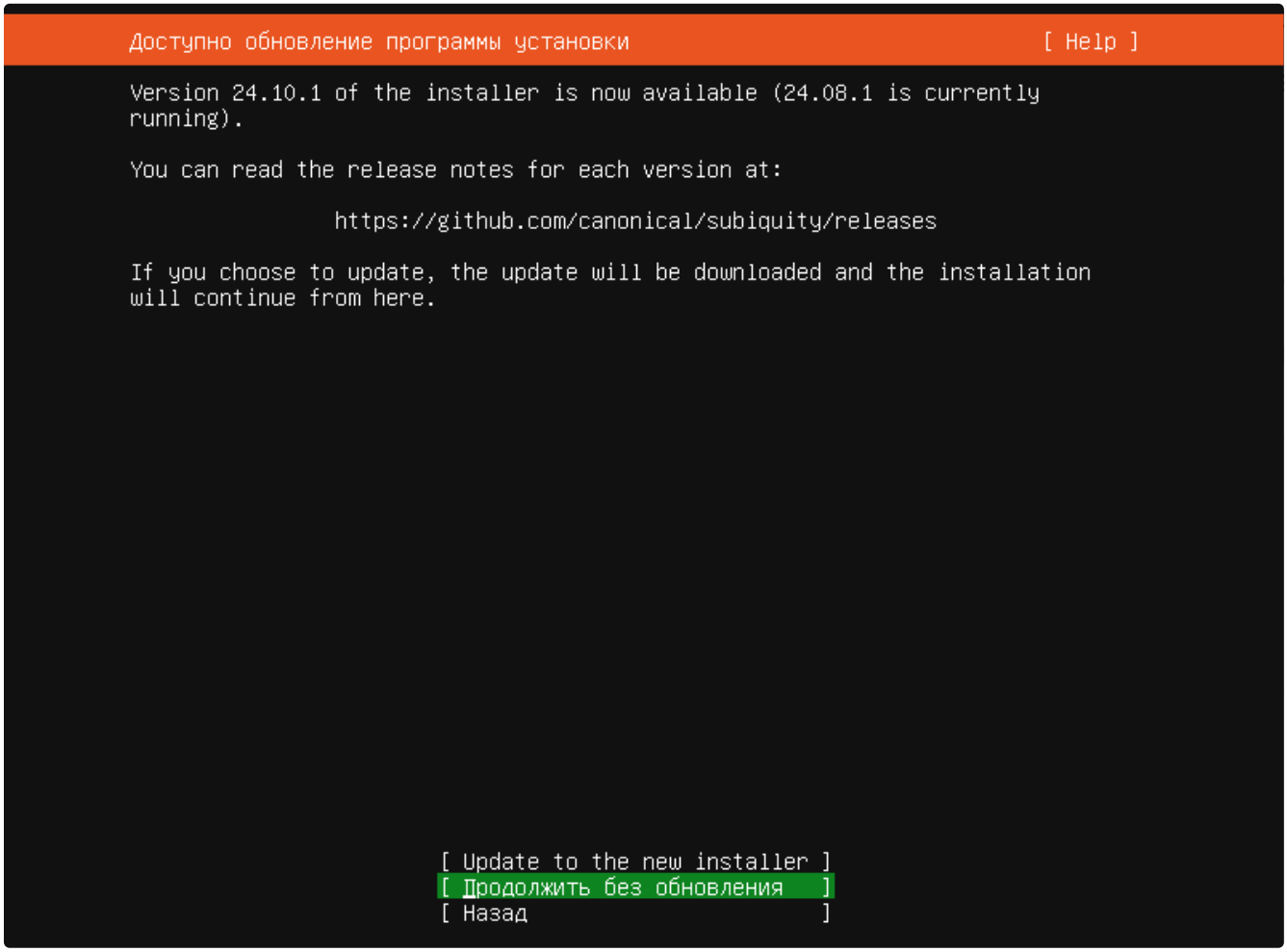
- После загрузки с установочного носителя выбрать "**Install Ubuntu Server**".
- Выбрать язык системы и раскладку клавиатуры.
- Настроить сетевой интерфейс для подключения к сети Интернет:
- Создать партиции на диске в соответствии с таблицей 1
- Настроить имена сервера и пользователя
- Установить OpenSSH server
- Перезагрузить успешно установленную ОС

Установка ОС Ubuntu Server 22.04

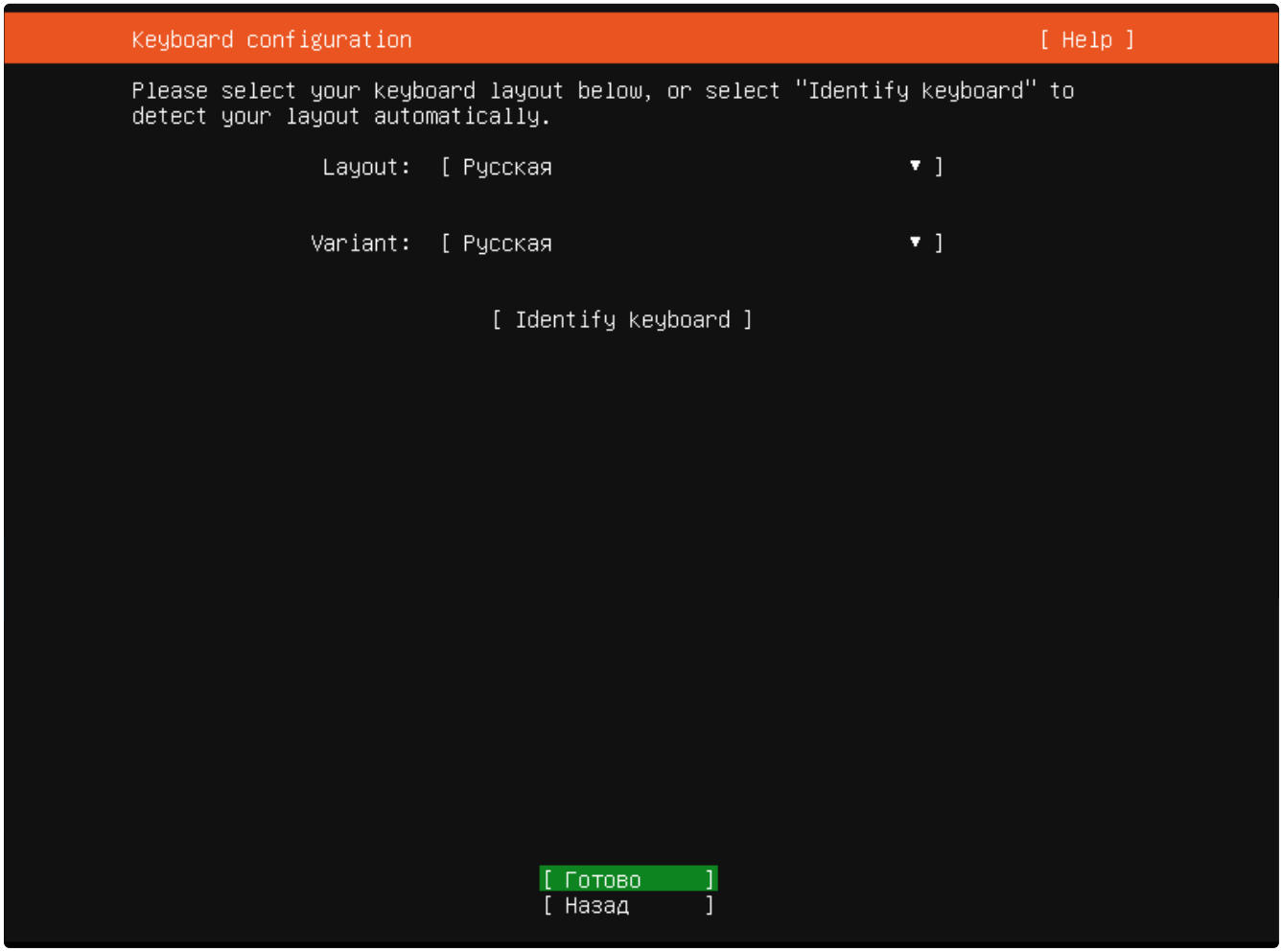
Выбираем язык интерфейса



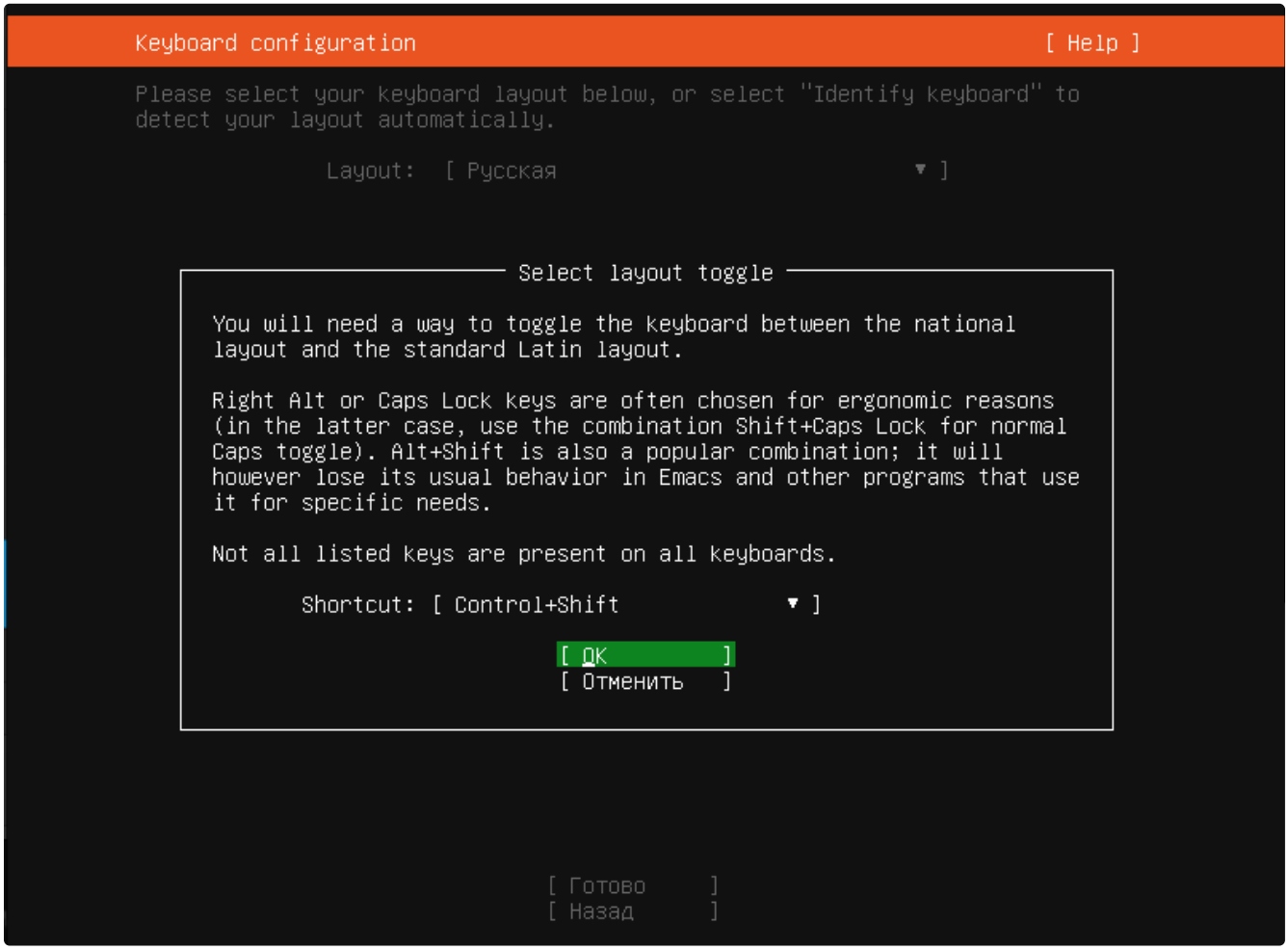
Продолжить без обновления



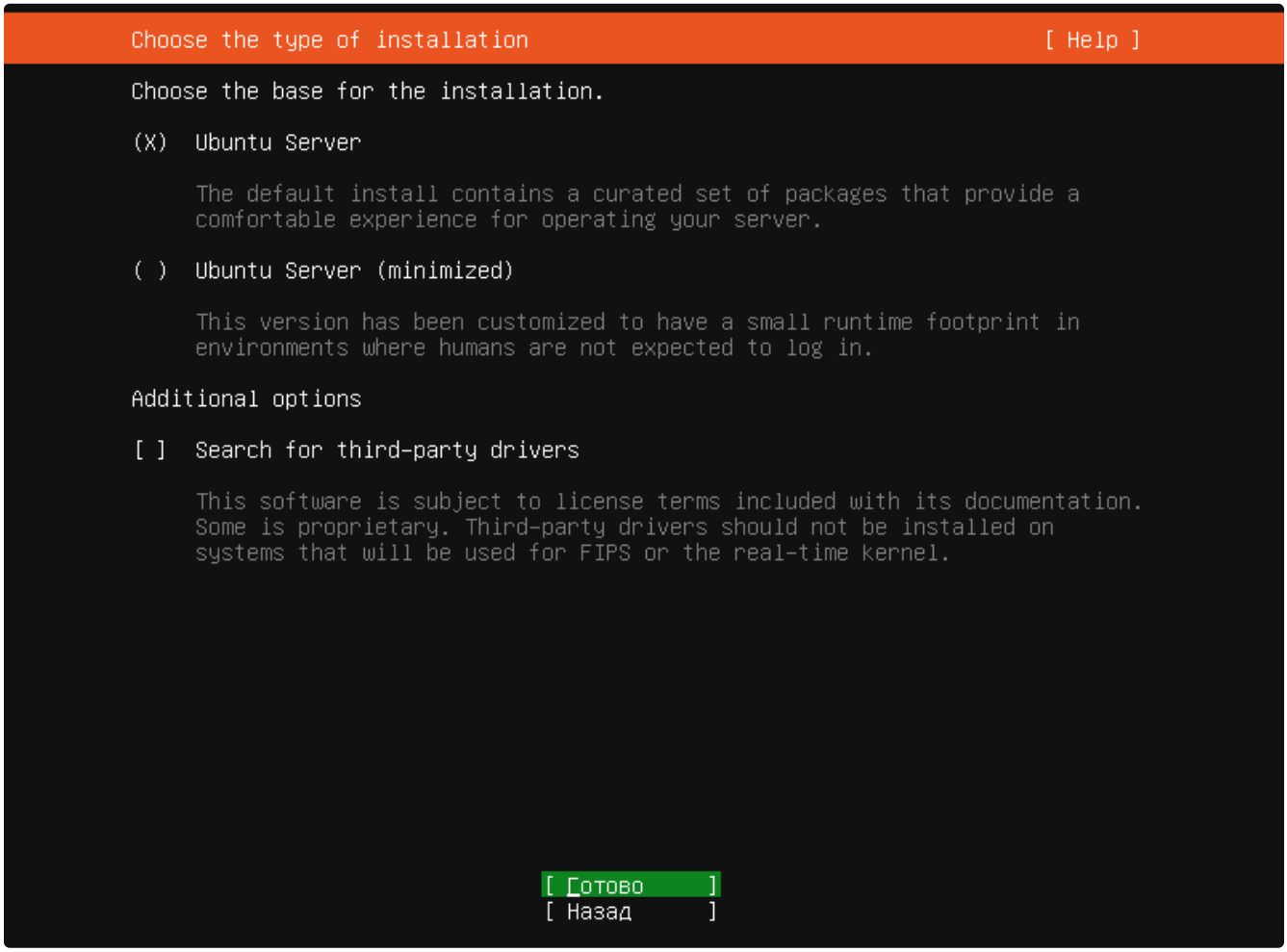
Выбираем раскладку клавиатуры



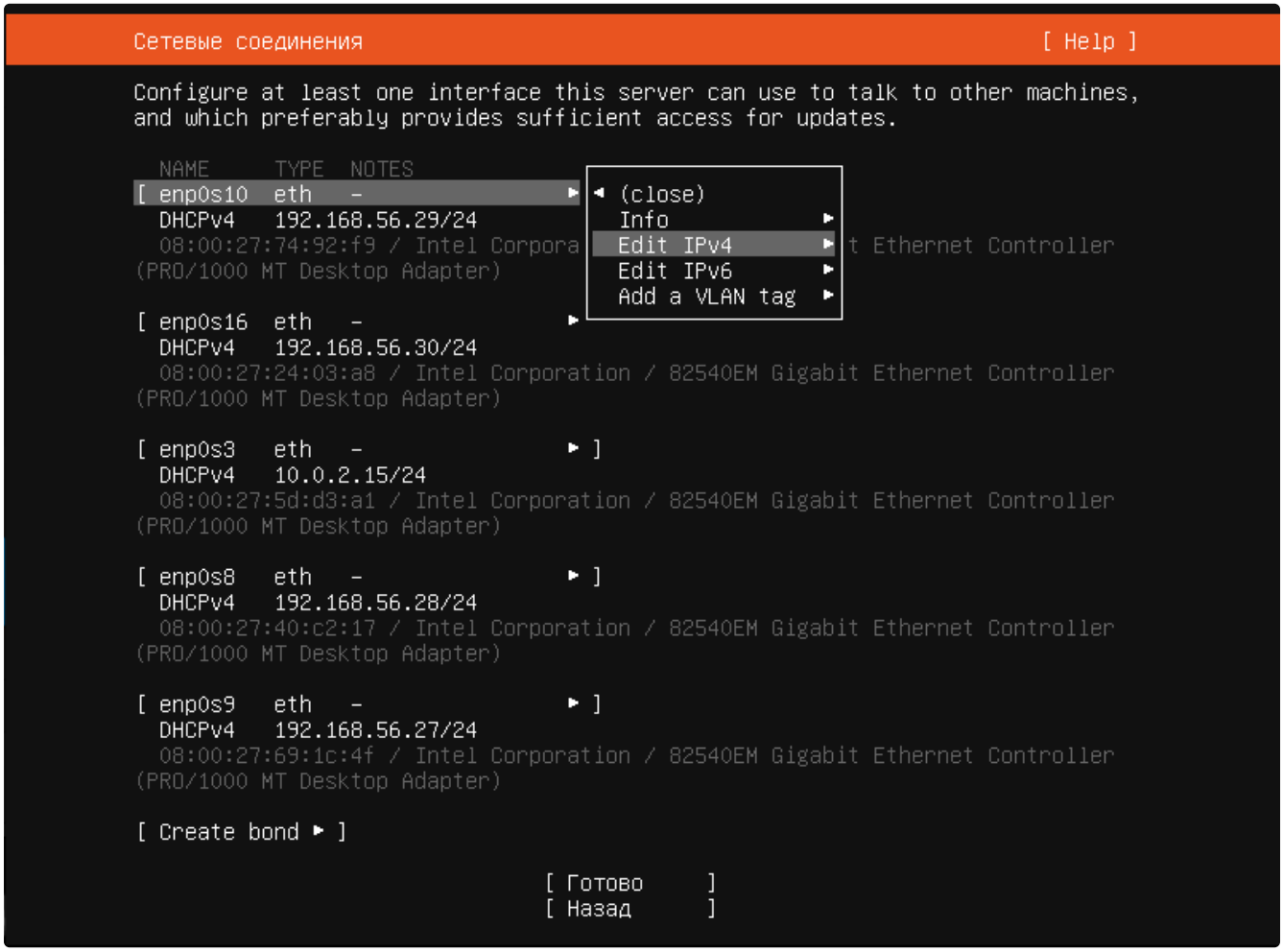
Выбираем переключатель раскладок Рус/Eng



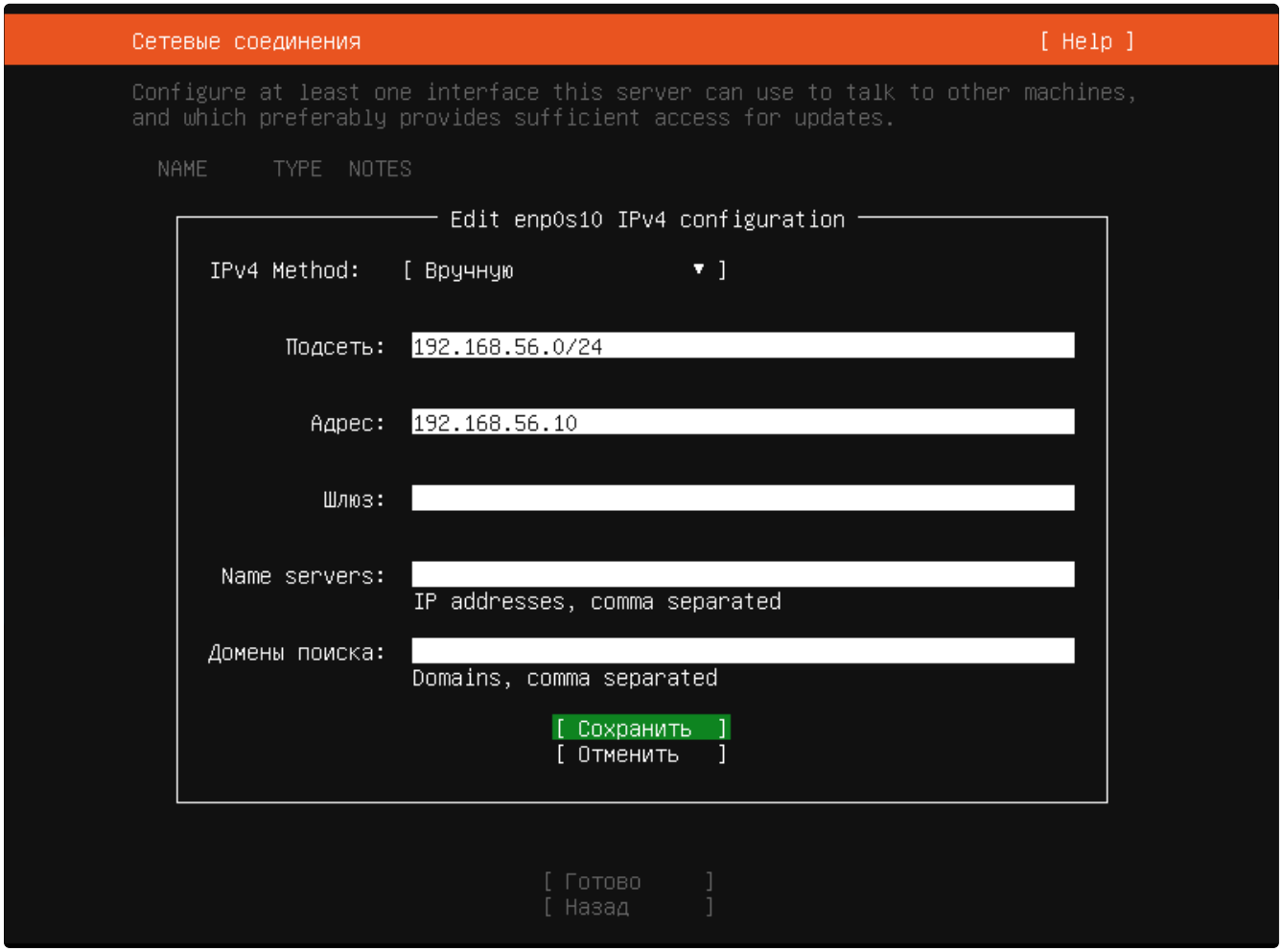
Выбираем стандартную установку Ubuntu Сервер



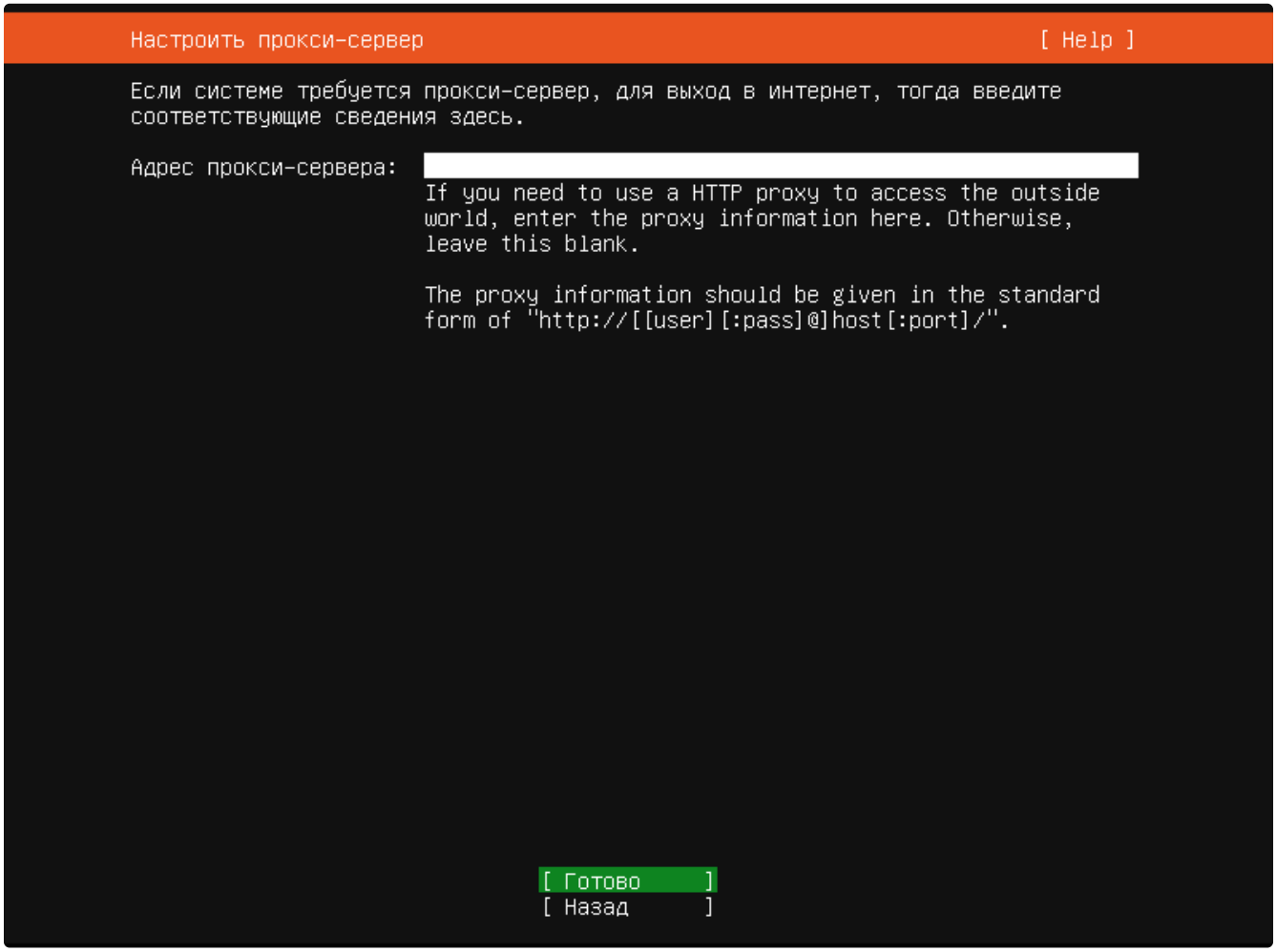
Выбираем Eht: интерфейс для установки Ubuntu (должен будет иметь доступ в интернет)



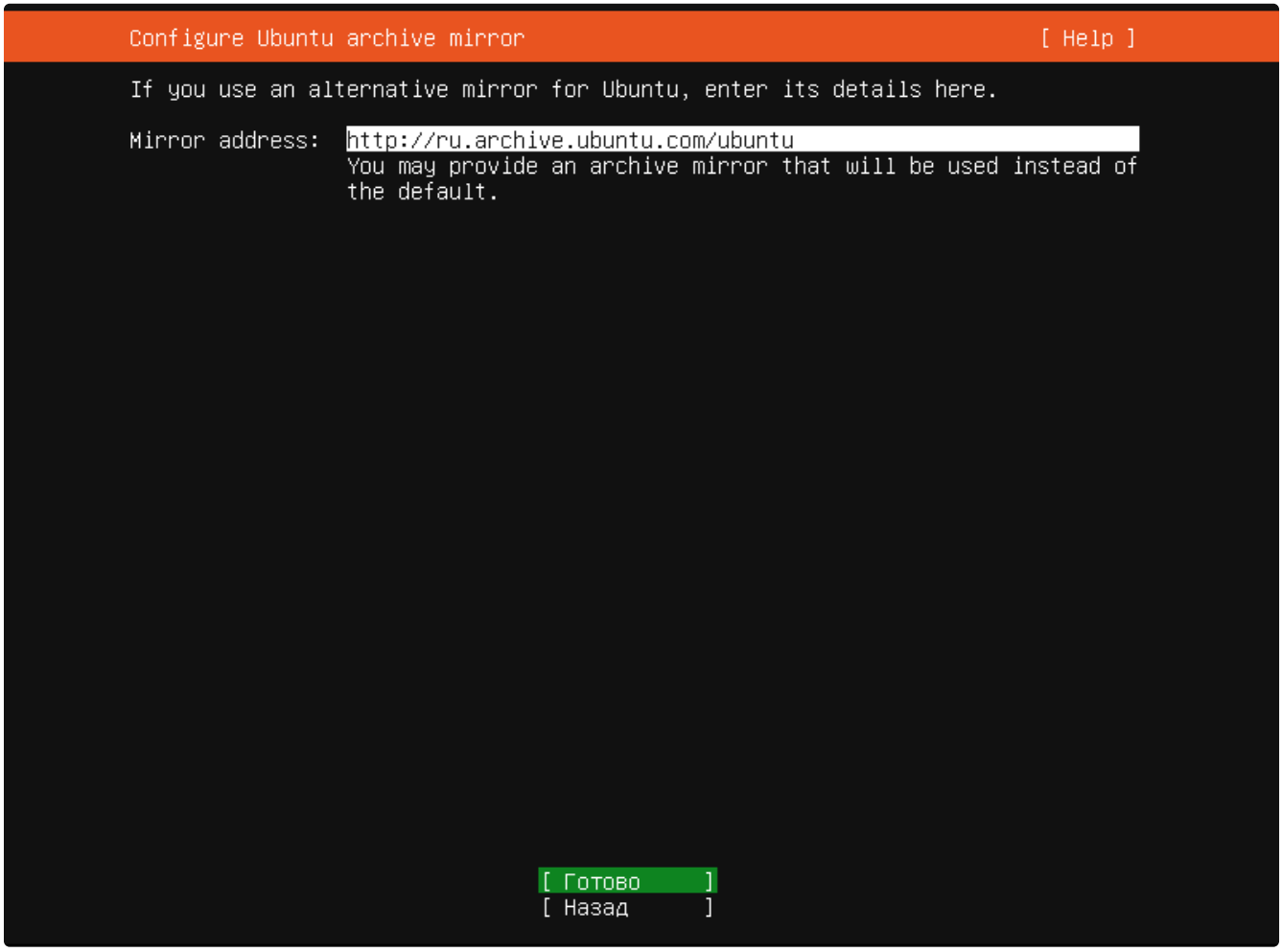
Установить IP-адрес для доступа в интернет(DHCP использовать нельзя)



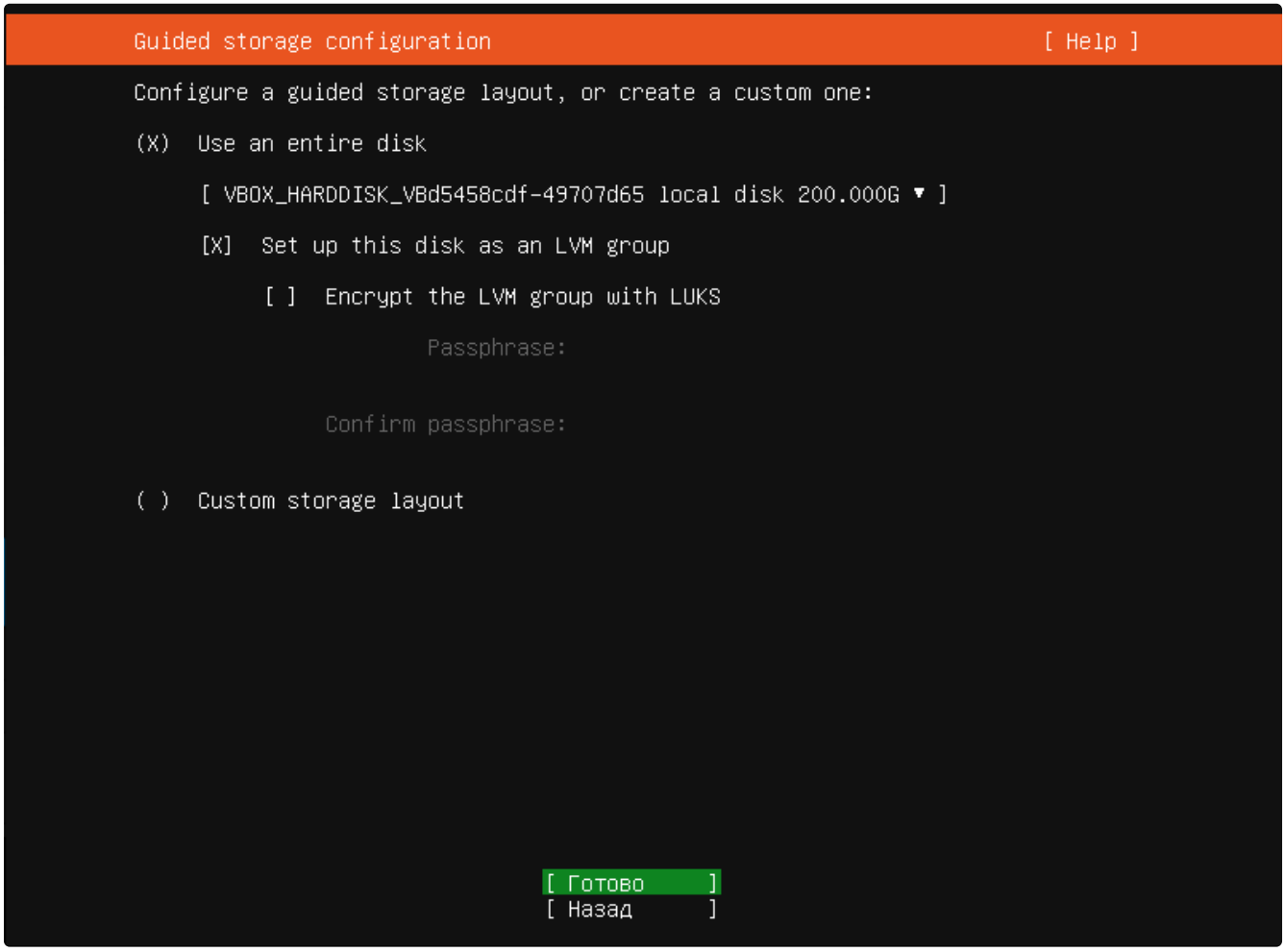
Прокси-сервер не используем(если в вашем проекте применяется то укажите)



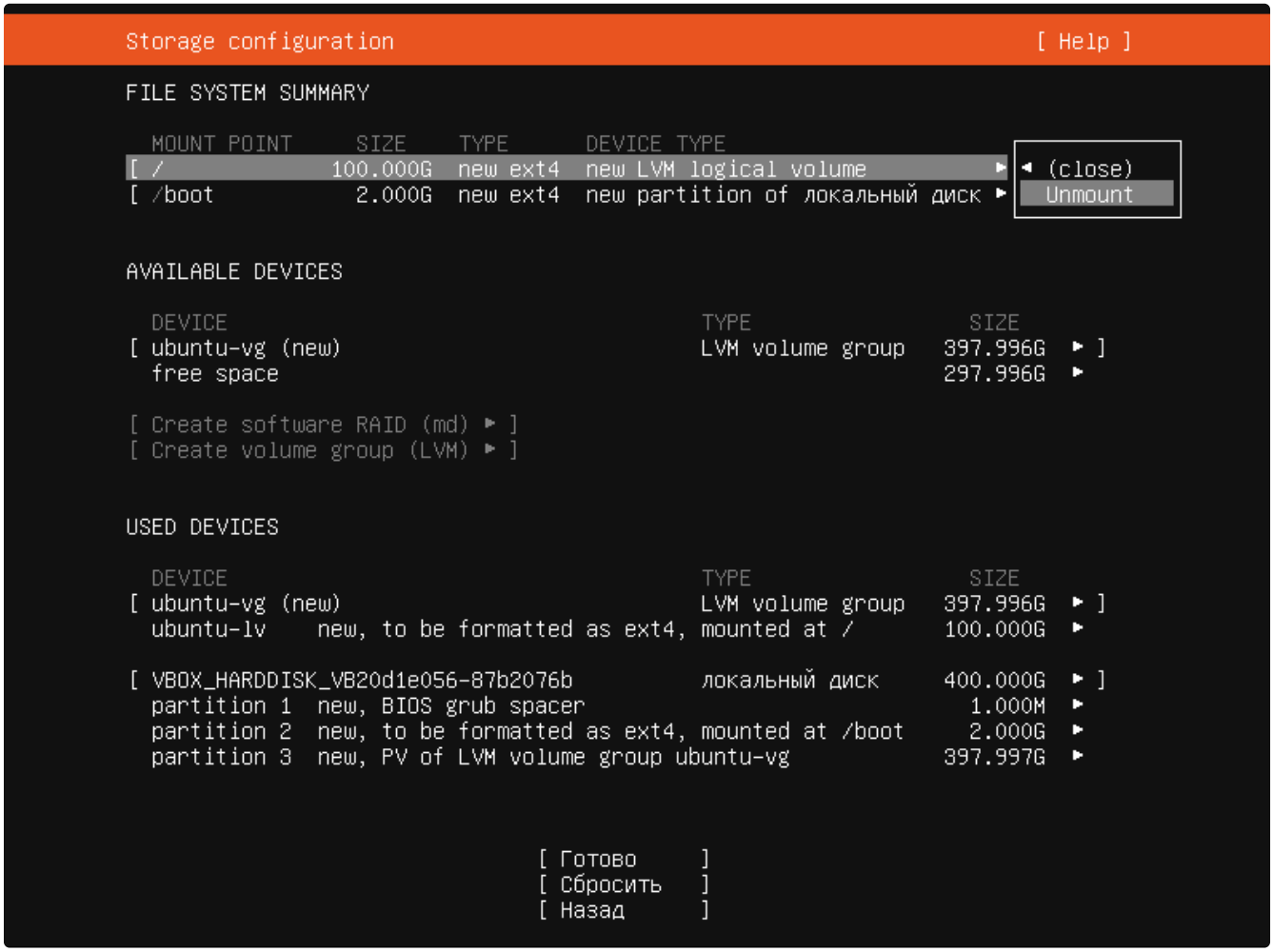
Значение по умолчанию



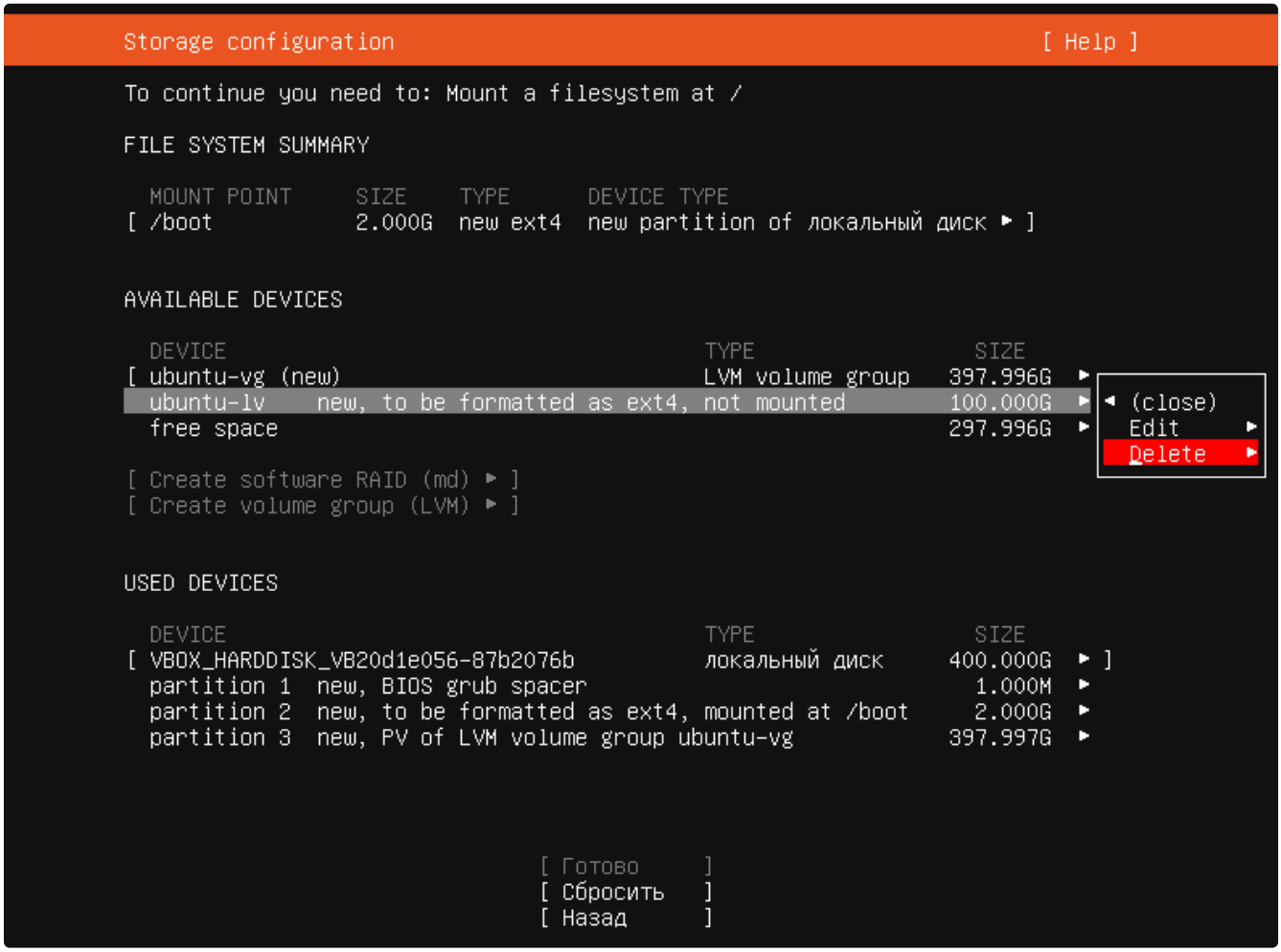
Создаем партиции на диске , используем LVM



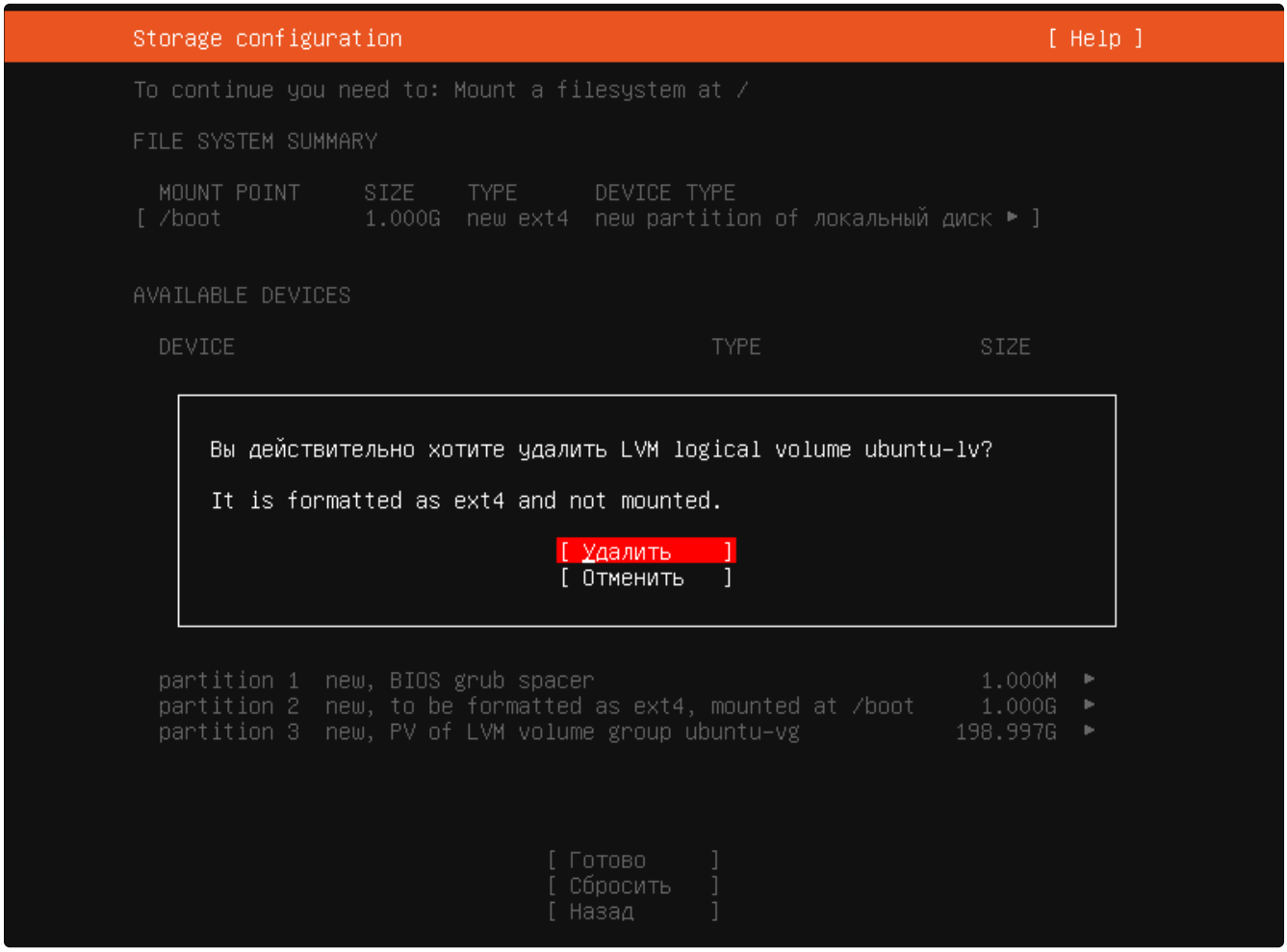
По умолчанию созданы две партиции /boot и / , /boot оставляем , а / - удаляем. Вначале размонтировать



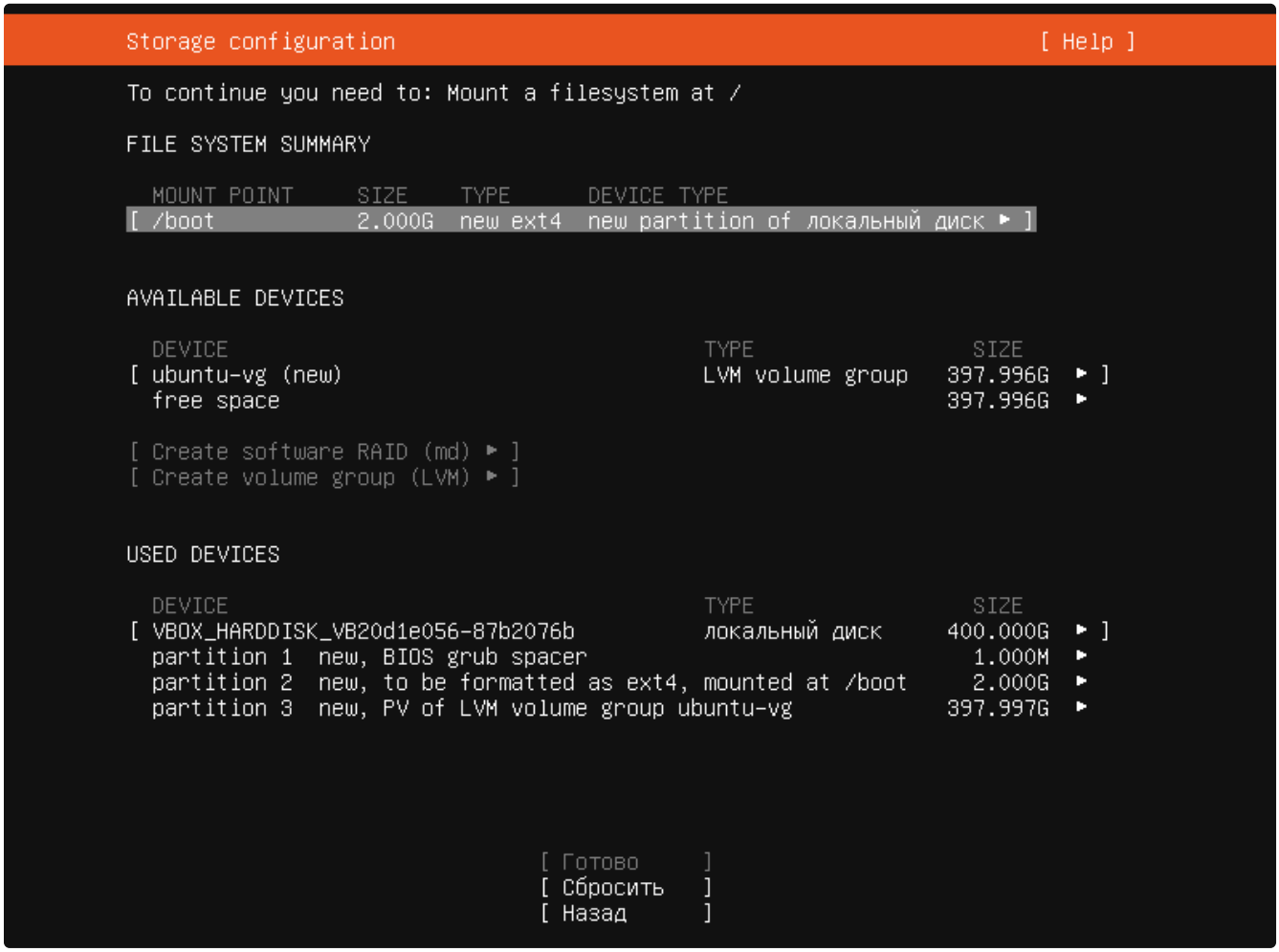
Затем удалить.



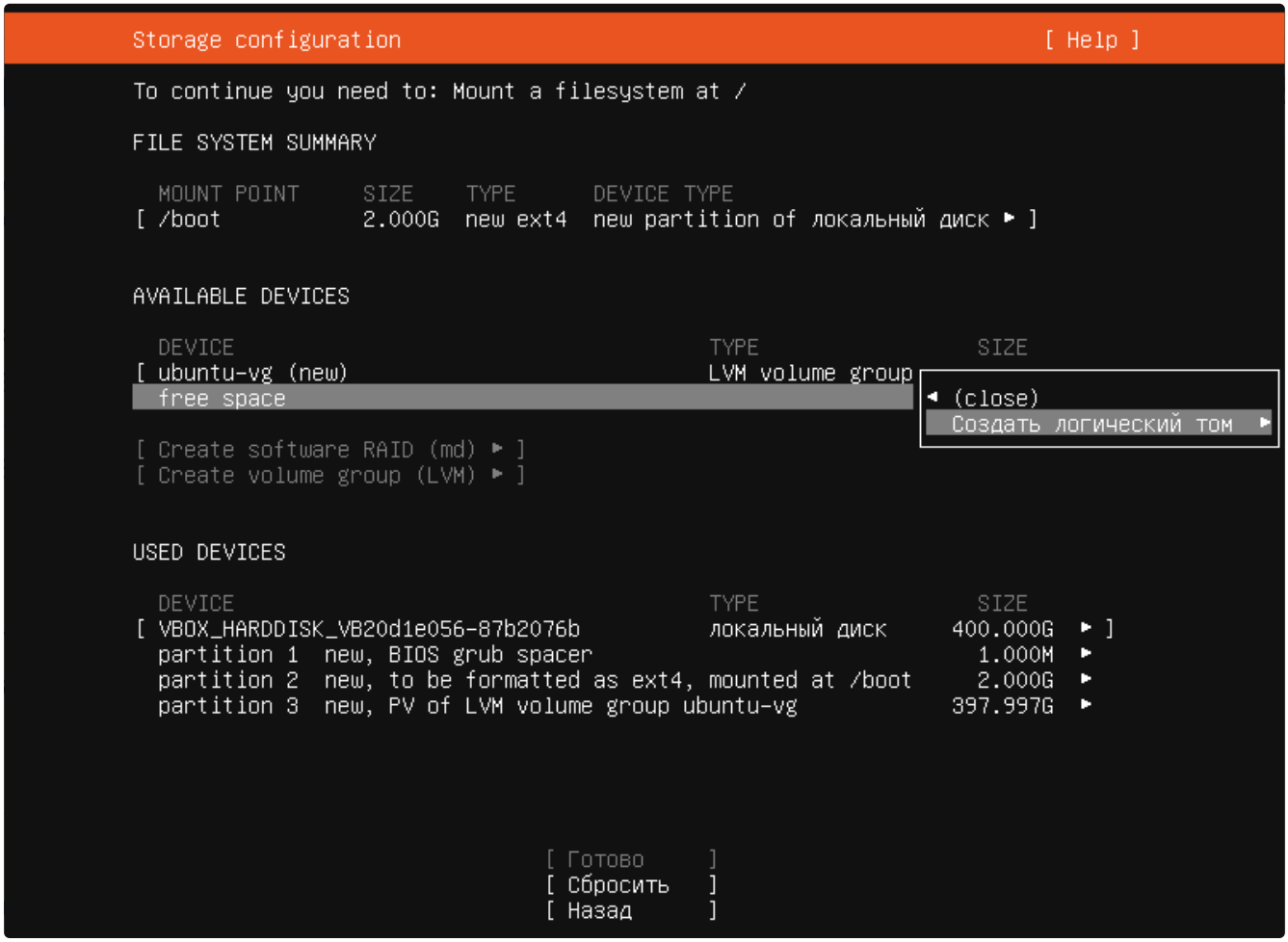
Подтвердить



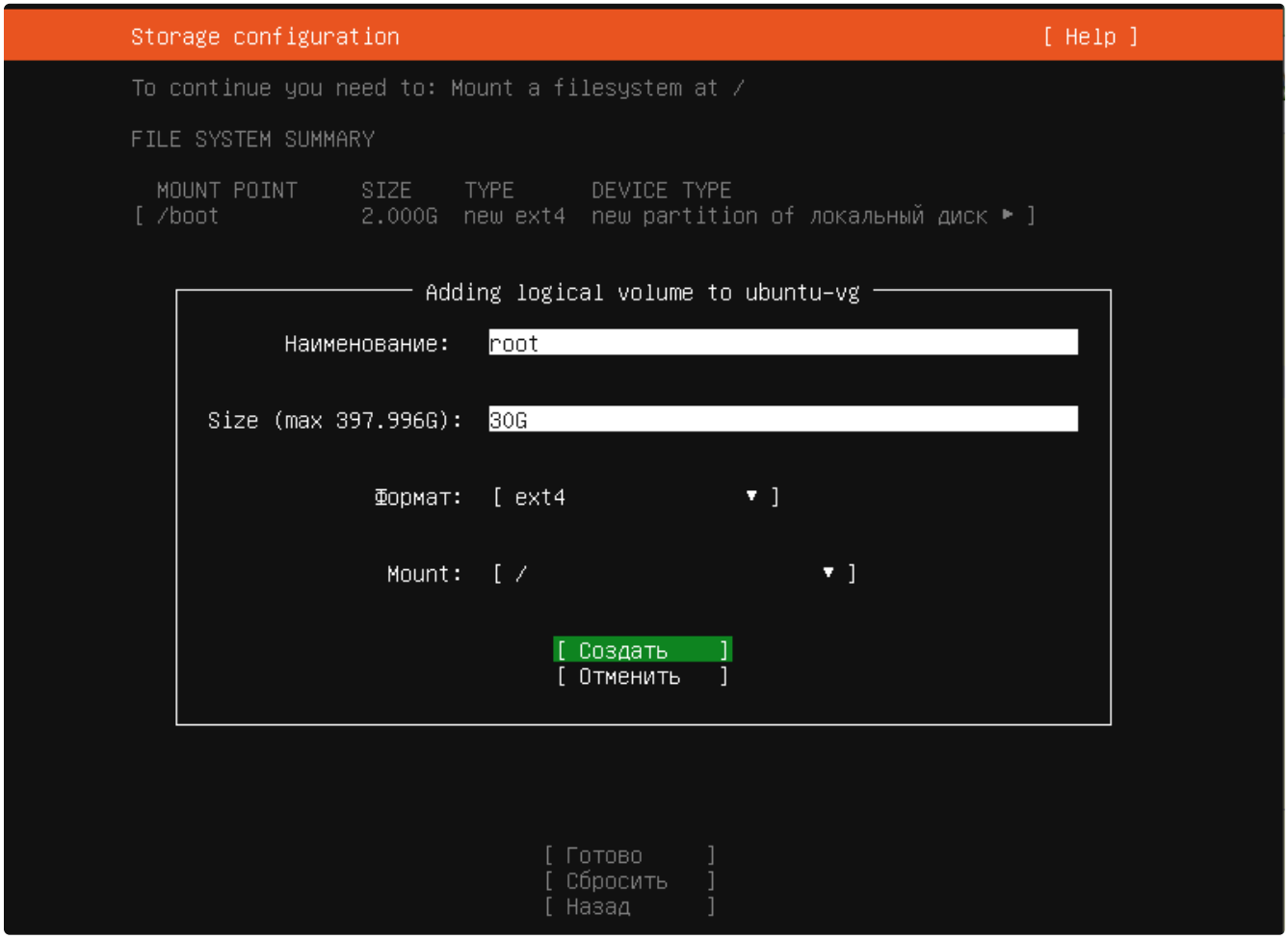
Оставить только /boot



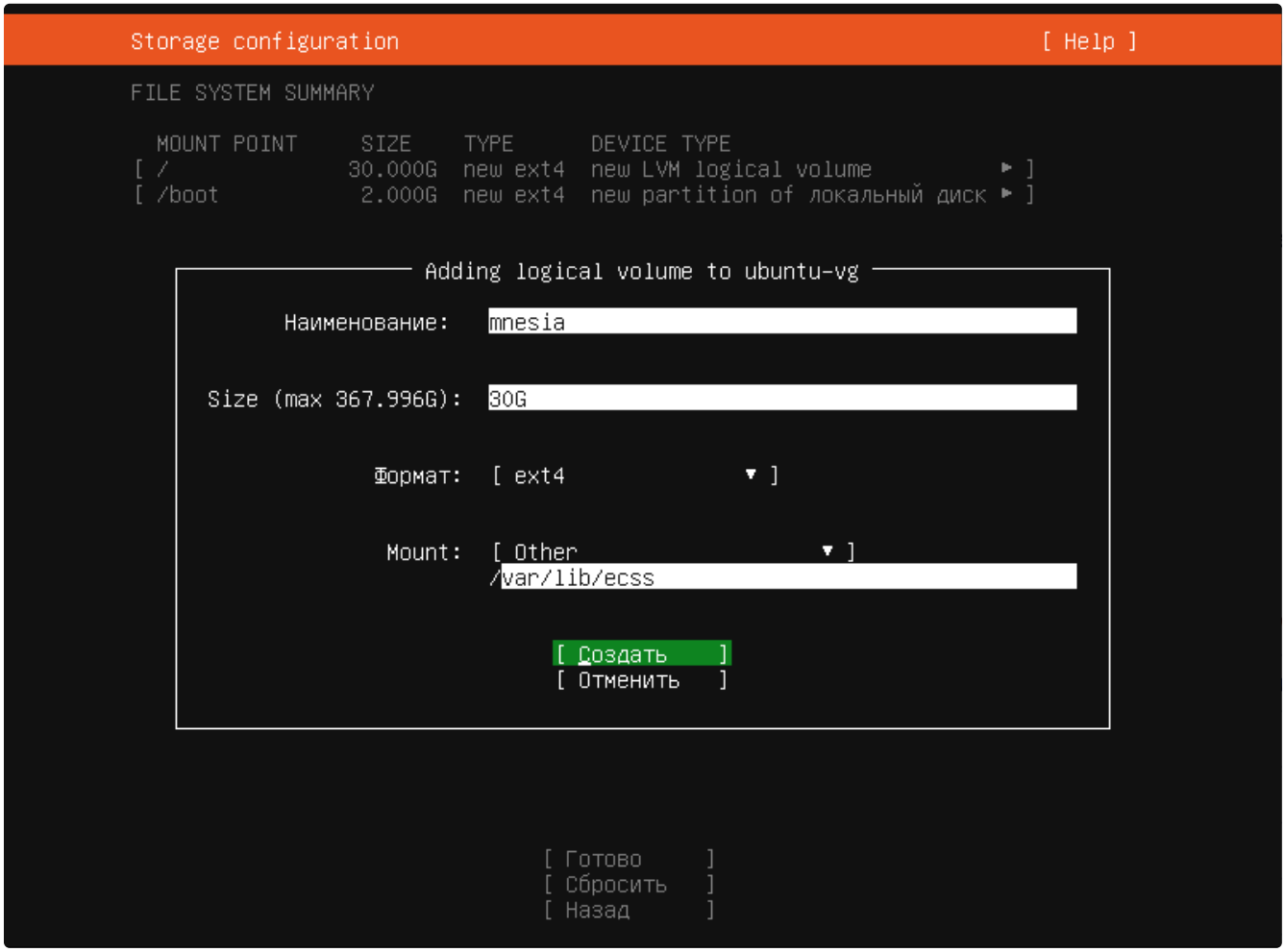
Создать партиции в соответствии с размерностью указанной в таблице ниже



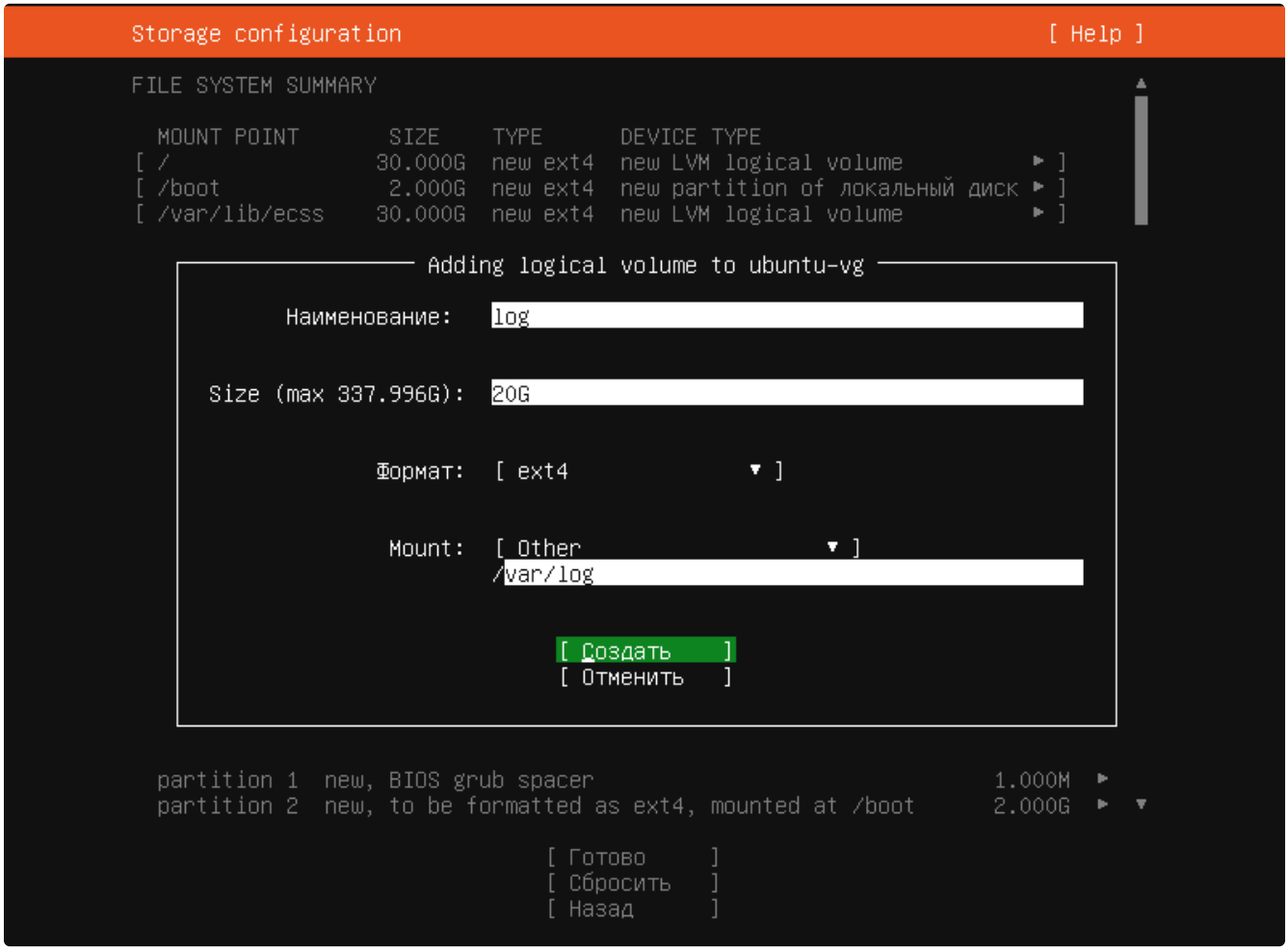
root размер 30 Гб , точка монтирования /



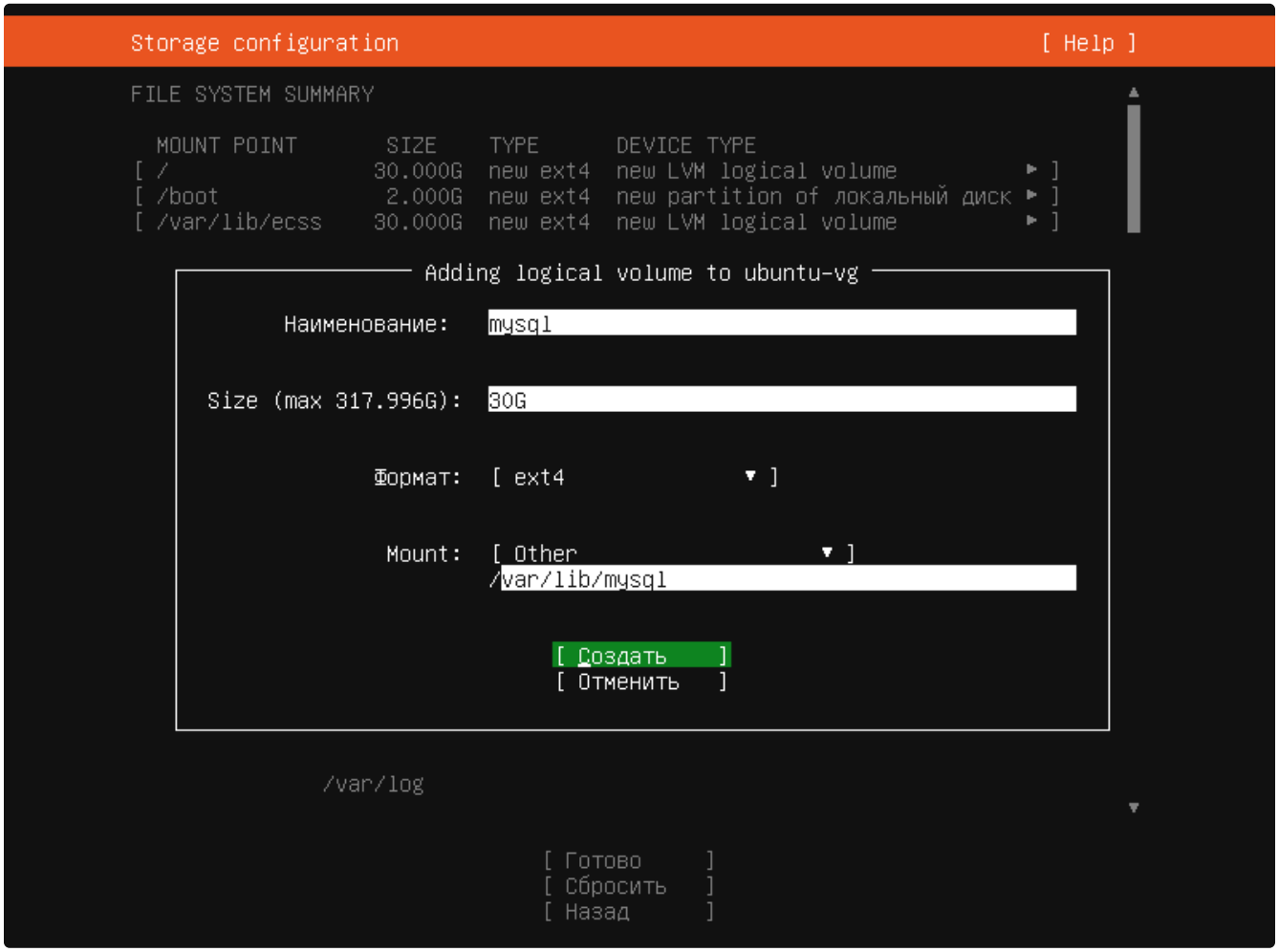
mnesia размер 30 Гб, точка монтирования /var/lib/ecss



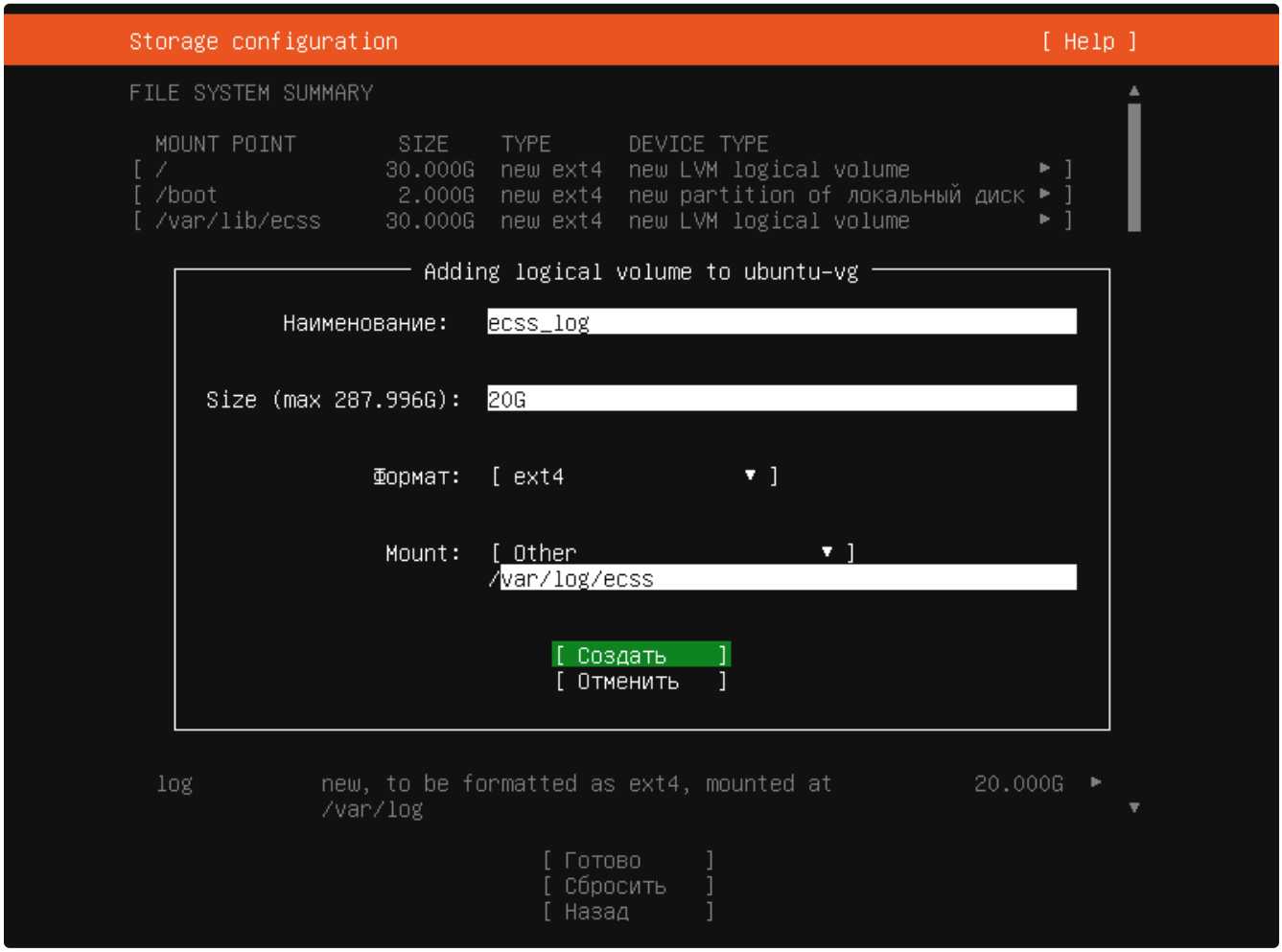
log размер 20 Гб, точка монтирования /var/log



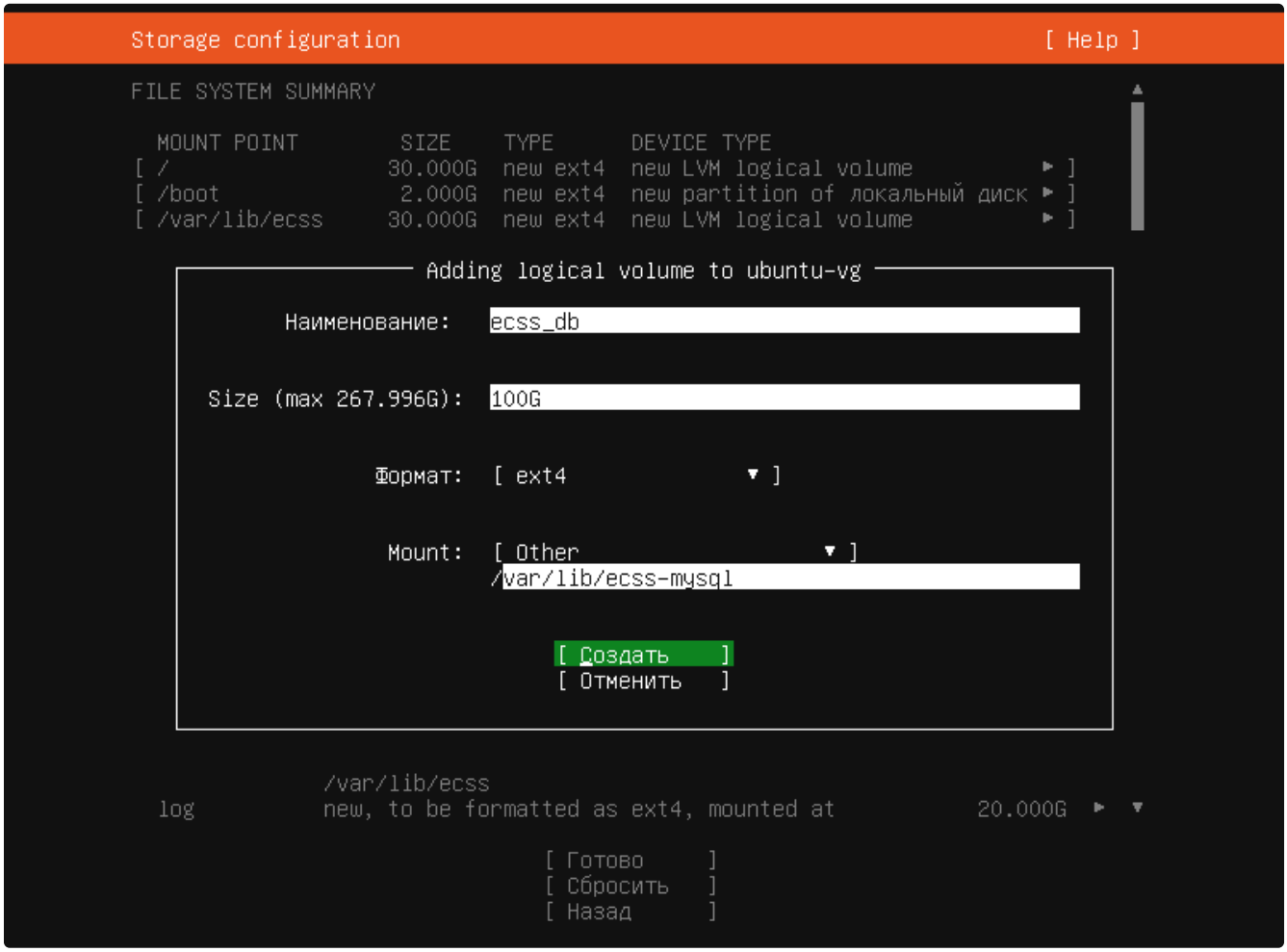
mysql, размер 30G, точка монтирования - /var/lib/mysql



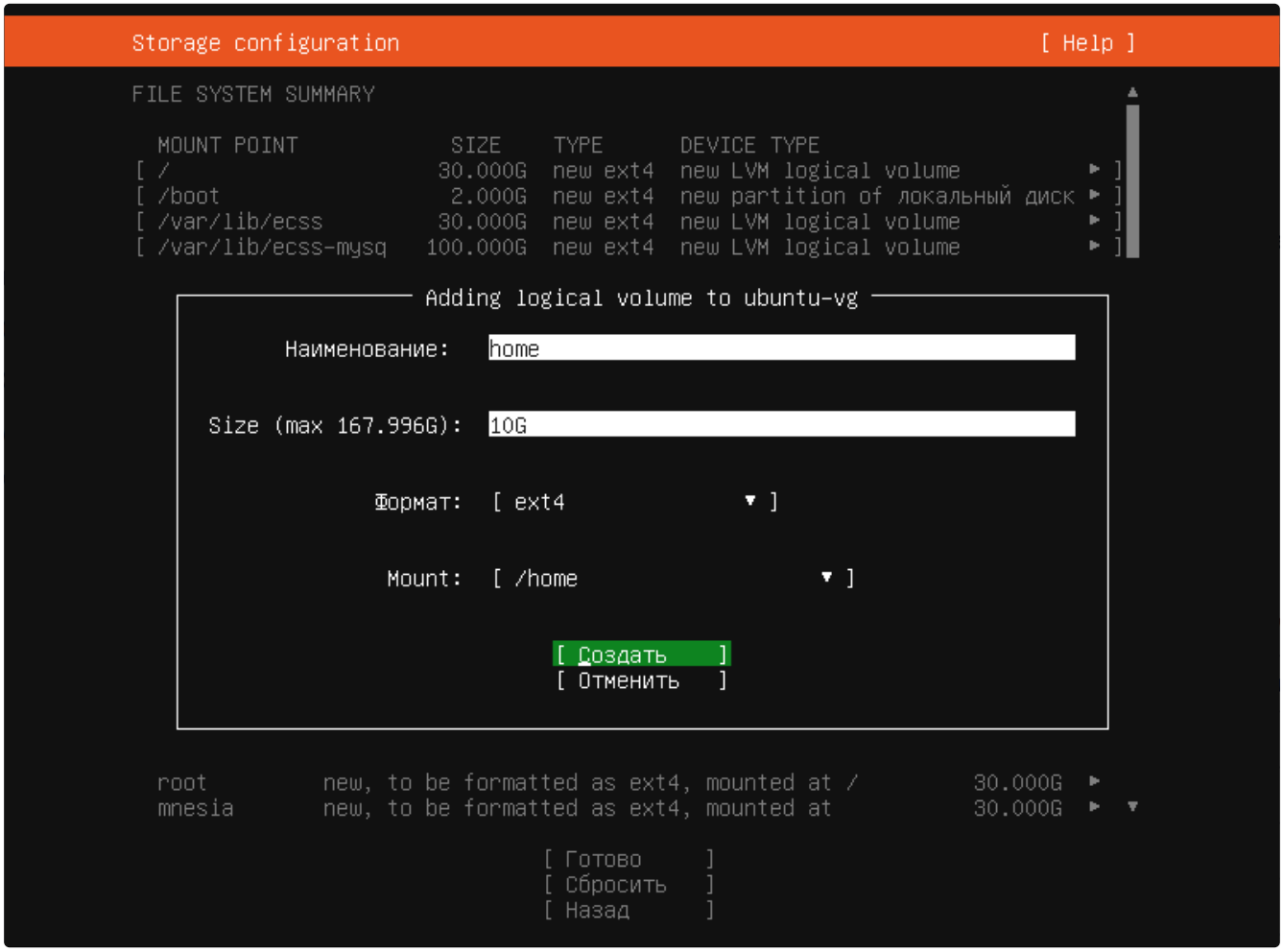
ecss_log, размер 20 Гб , точка монтирования /var/log/ecss



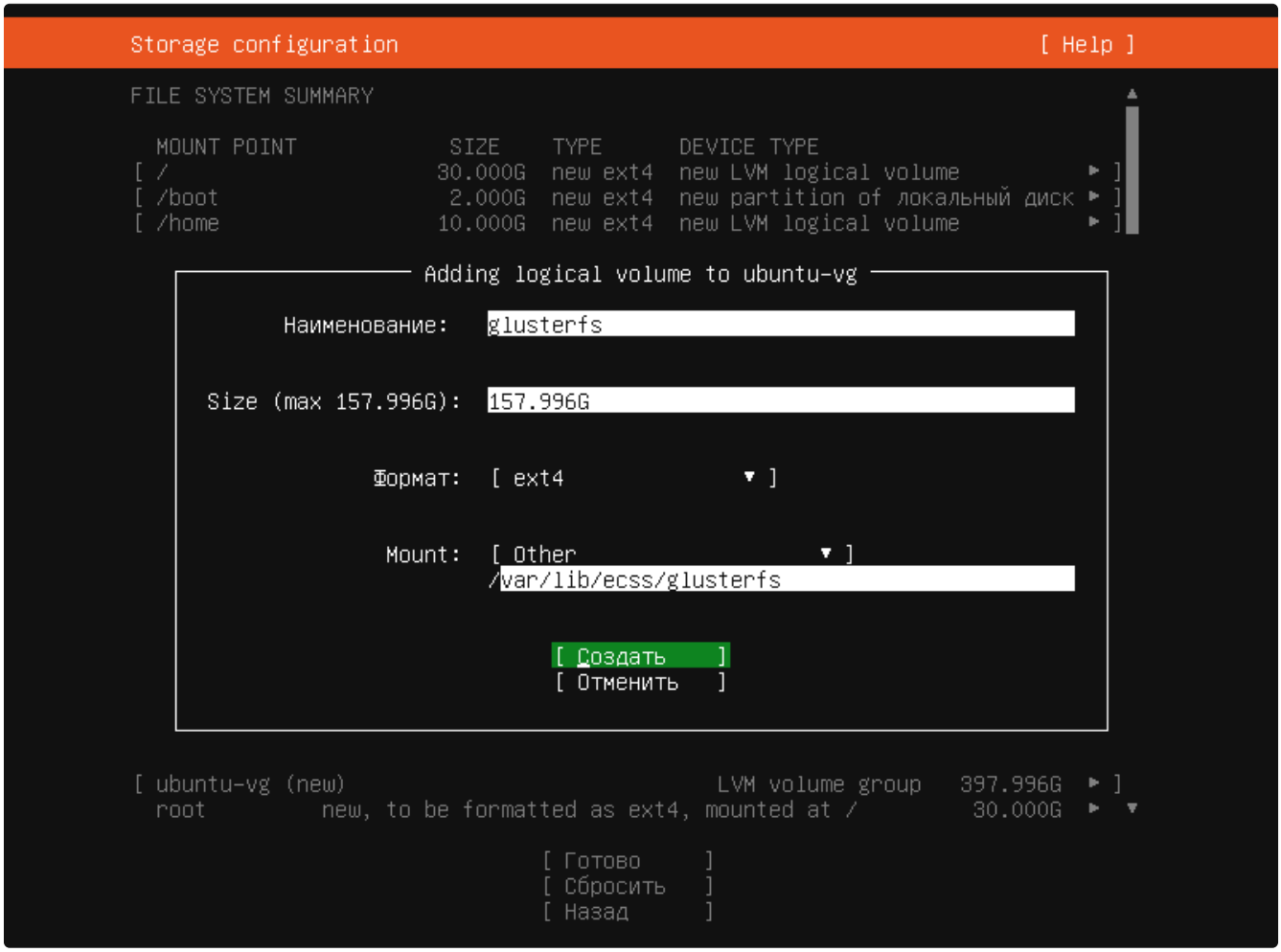
ecss_db, размер 100 Гб , точка монтирования - /var/lib/ecss-mysql



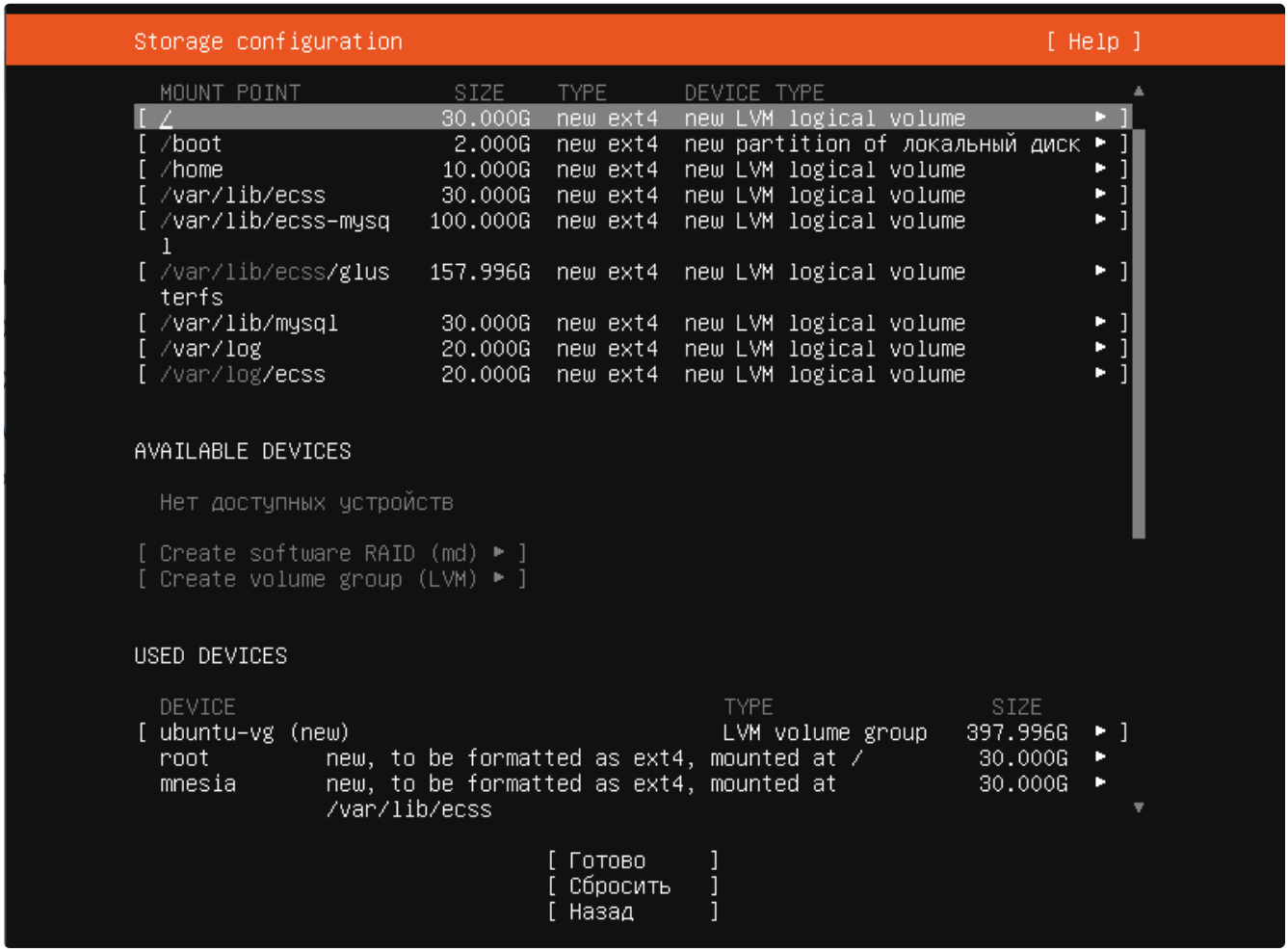
home, размер 10 Гб, точка монтирования - /home

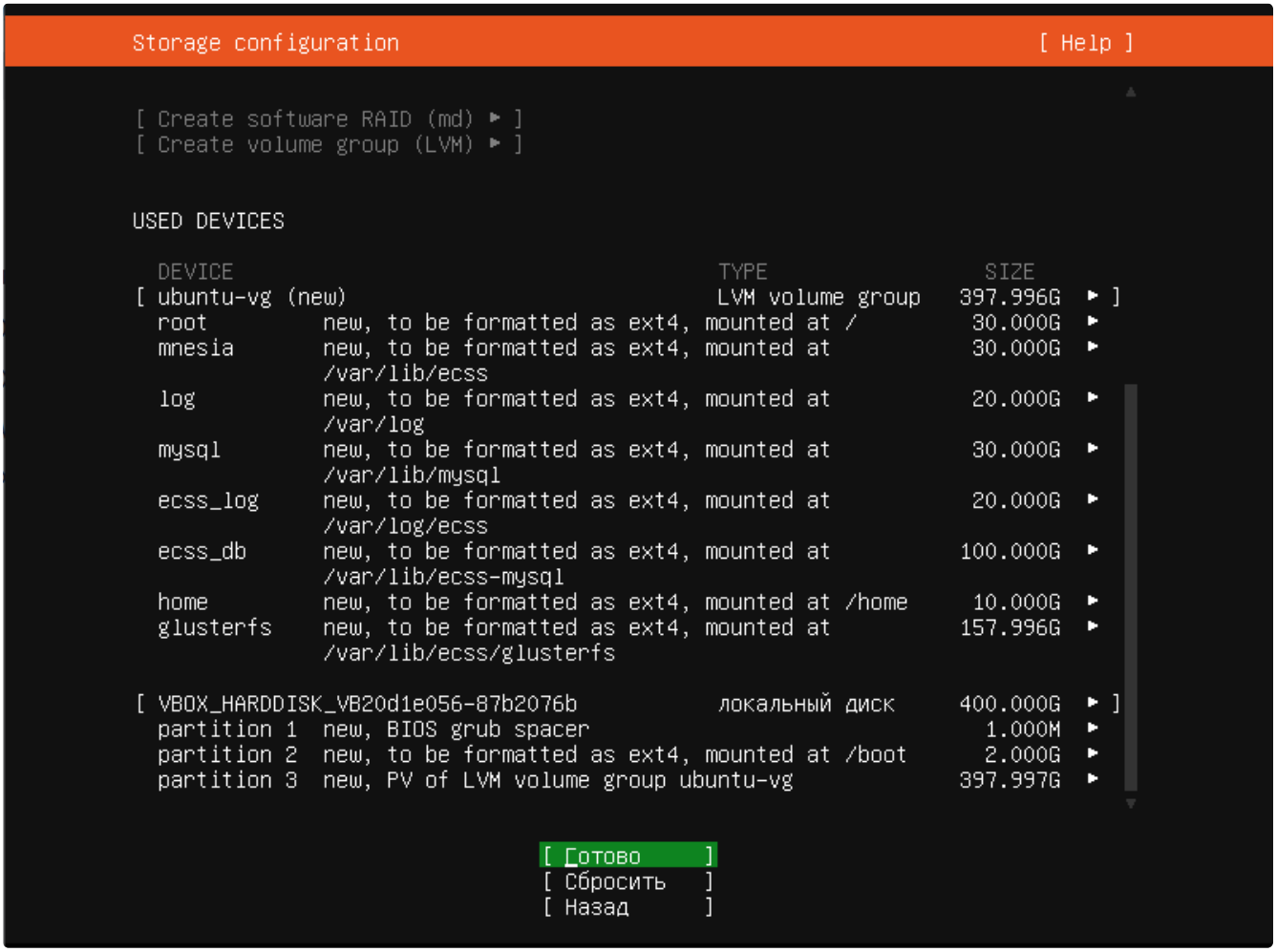


glusterfs - для кластера / для одного хоста - restfs , размер - все оставшееся место на диске, точка монтирования - /var/lib/ecss/glusterfs - для кластера , и /var/lib/ecss/restfs - для одного сервера

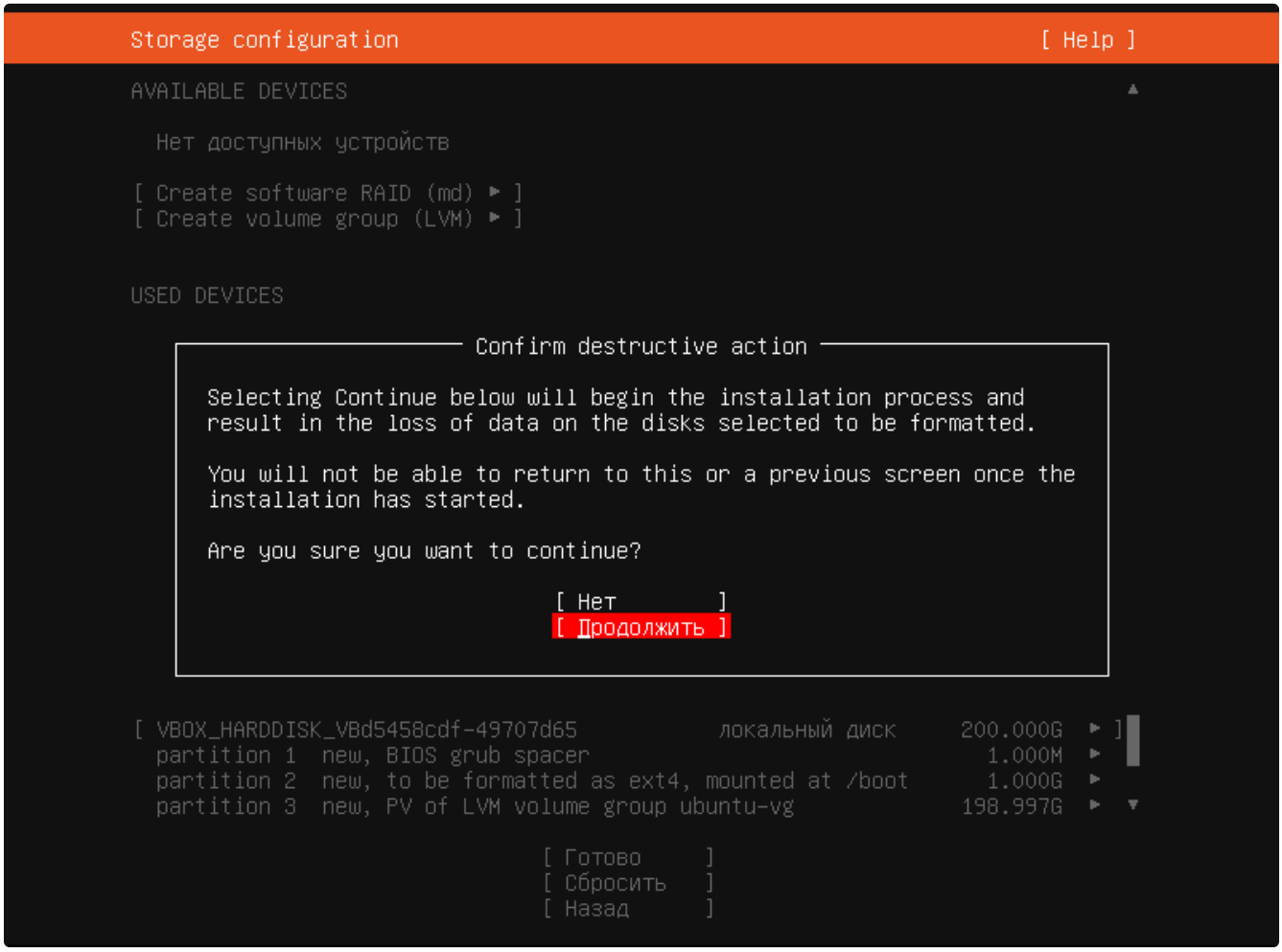


полученная конфигурация

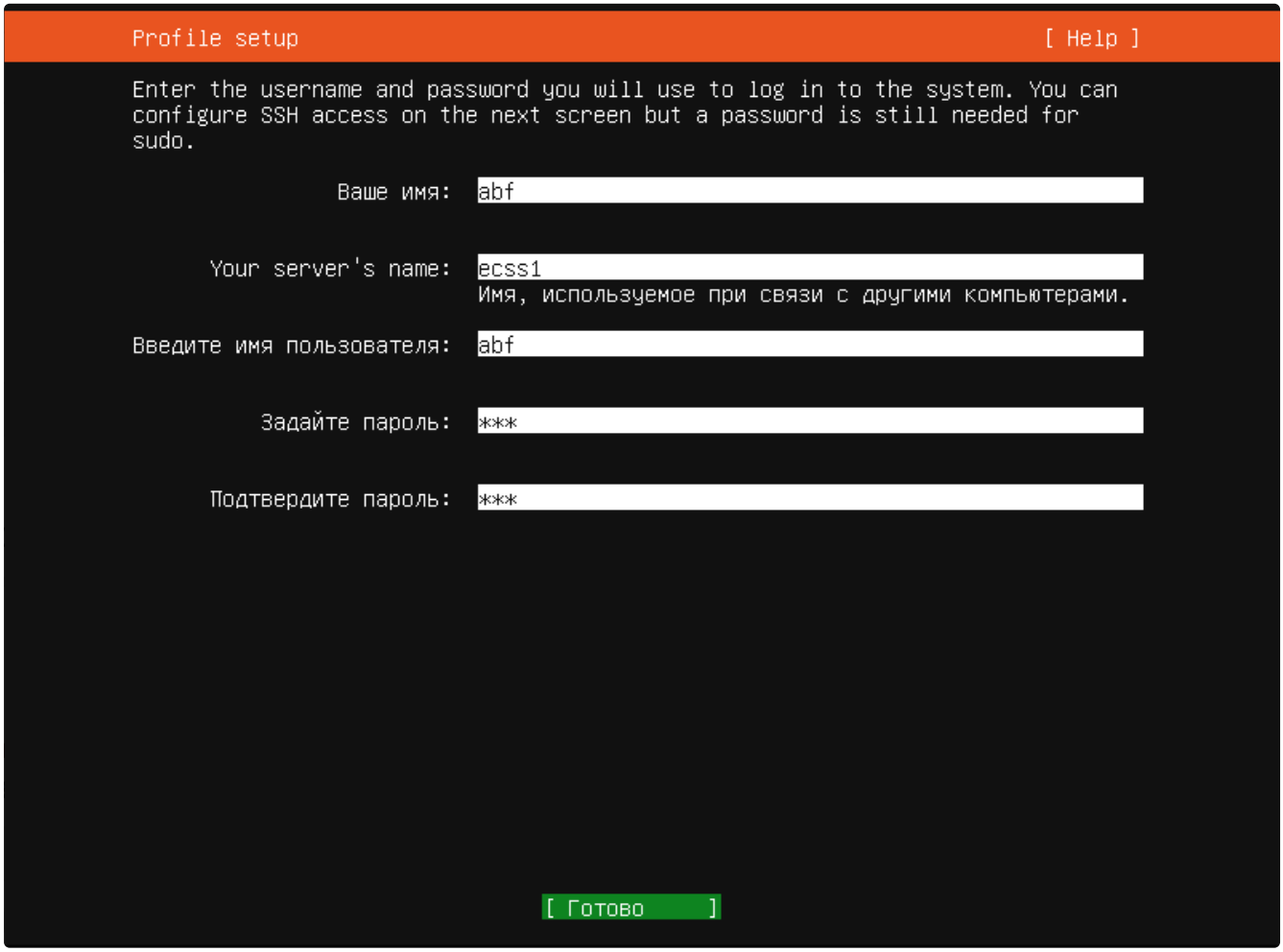

Storage configuration
[Help]
MOUNT POINT SIZE TYPE DEVICE TYPE
[/ 30.000G new ext4 new LVM logical volume ▸]
[/boot 2.000G new ext4 new partition of локальный диск ▸]
[/home 10.000G new ext4 new LVM logical volume ▸]
[/var/lib/ecss 30.000G new ext4 new LVM logical volume ▸]
[/var/lib/ecss-mysq 100.000G new ext4 new LVM logical volume ▸]
l
[/var/lib/ecss/glus 157.996G new ext4 new LVM logical volume ▸]
terfs
[/var/lib/mysql 30.000G new ext4 new LVM logical volume ▸]
[/var/log 20.000G new ext4 new LVM logical volume ▸]
[/var/log/ecss 20.000G new ext4 new LVM logical volume ▸]
AVAILABLE DEVICES
Нет доступных устройств
[Create software RAID (md) ▸]
[Create volume group (LVM) ▸]
USED DEVICES
DEVICE TYPE SIZE
[ubuntu-vg (new) LVM volume group 397.996G ▸]
root new, to be formatted as ext4, mounted at / 30.000G ▸
mnesia new, to be formatted as ext4, mounted at 30.000G ▸
/var/lib/ecss
[Готово]
[Сбросить]
[Назад]



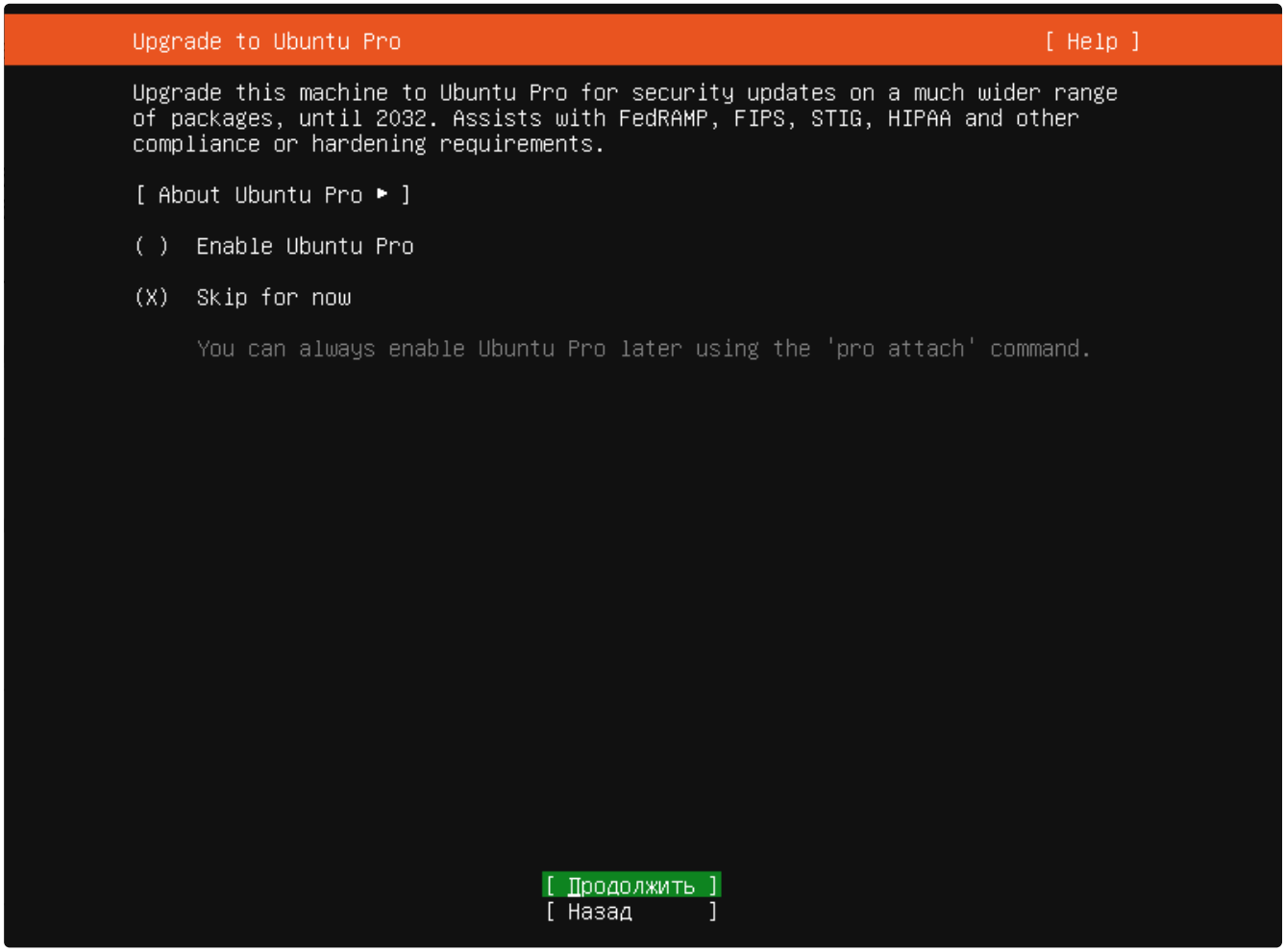
Подтвердить



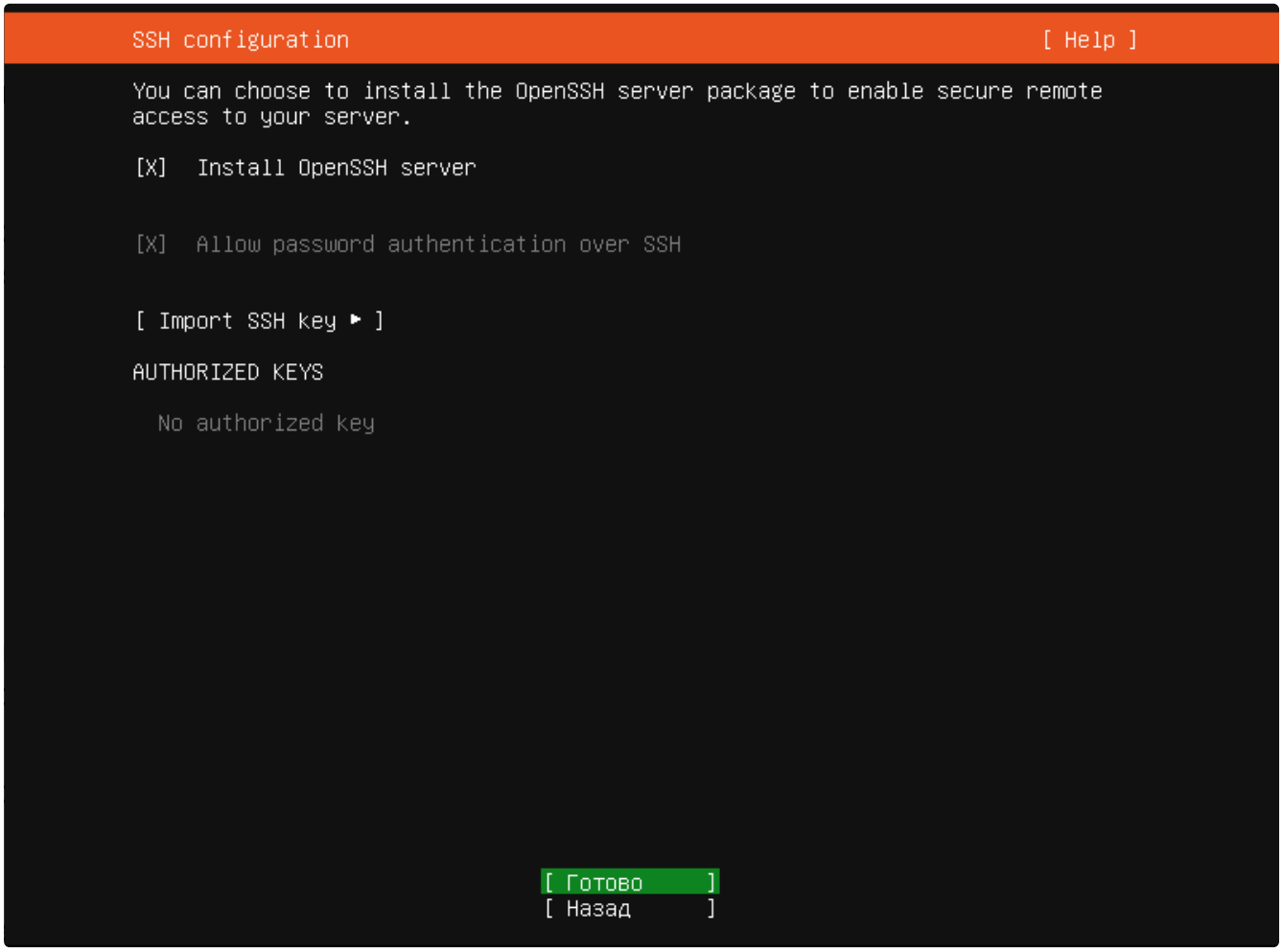
Создать пользователя и имя хоста, в ECSS10 рекомендуемое имя хоста ecss1/ecss2 (используется при интеграции лицензий), имя пользователя (в примере abf / пароль на ваше усмотрение)



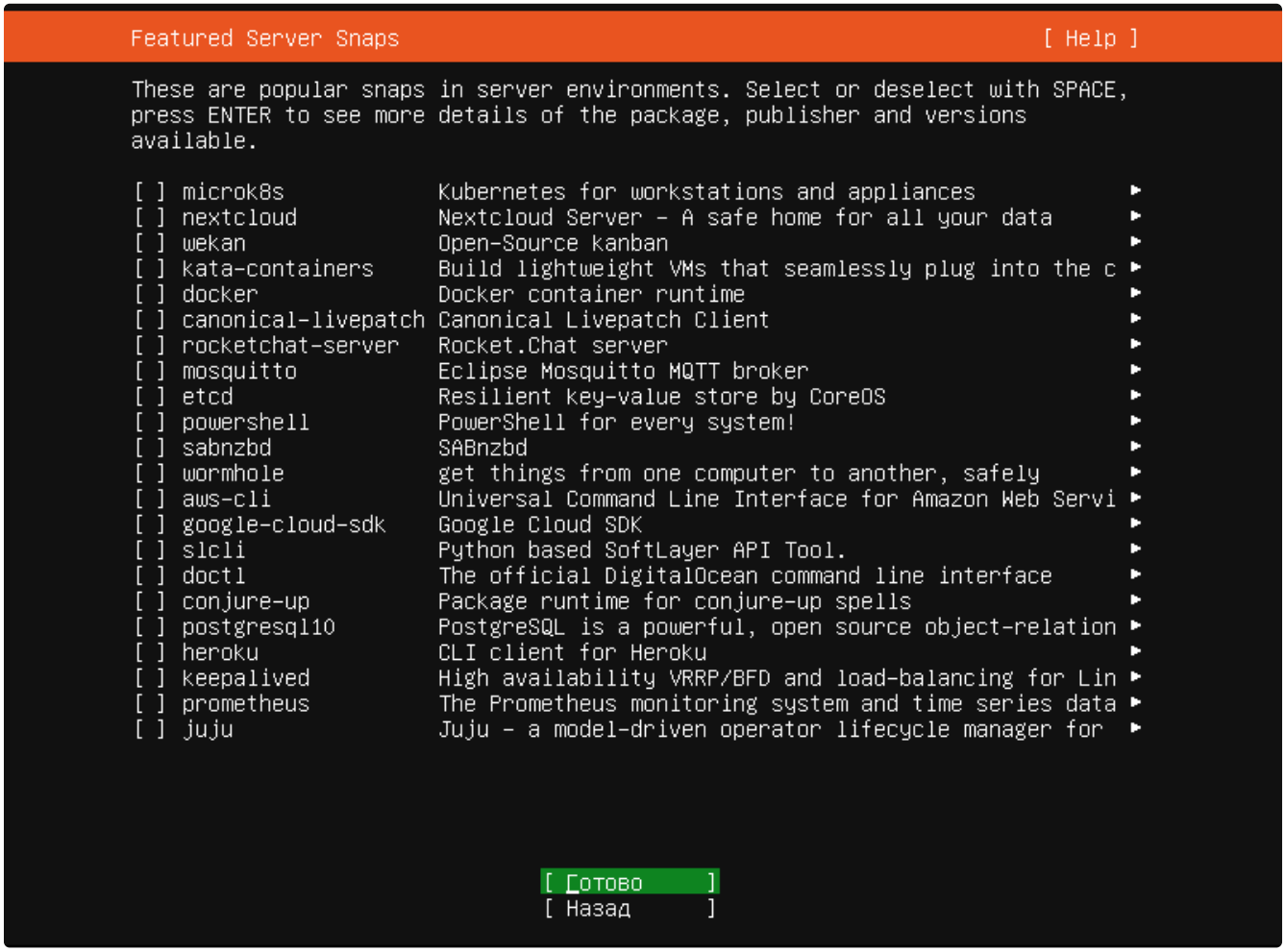
При установки выбираем Free установку:



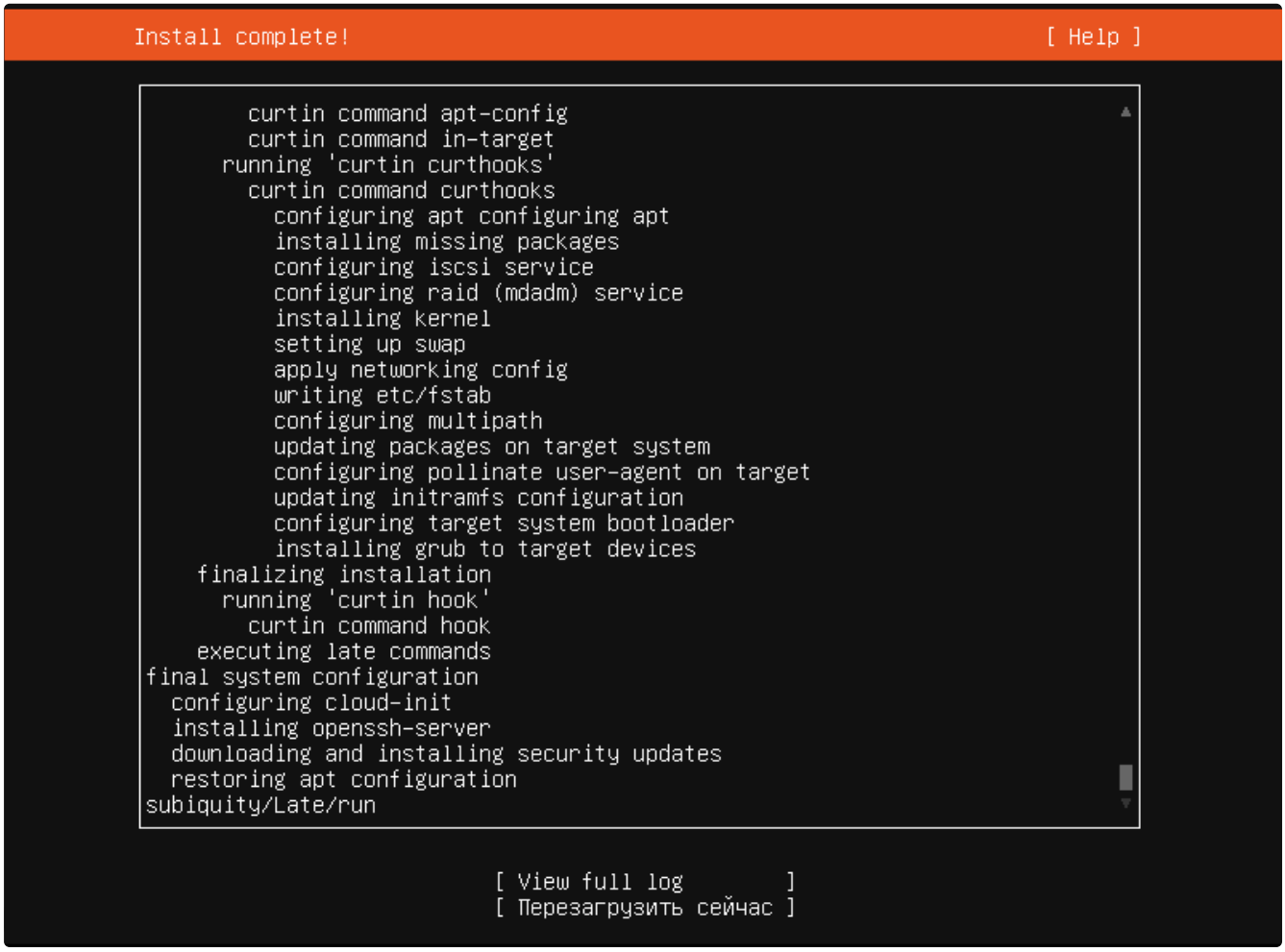
При инсталляции Ubuntu Linux, ставим openSSH, для удаленного управления



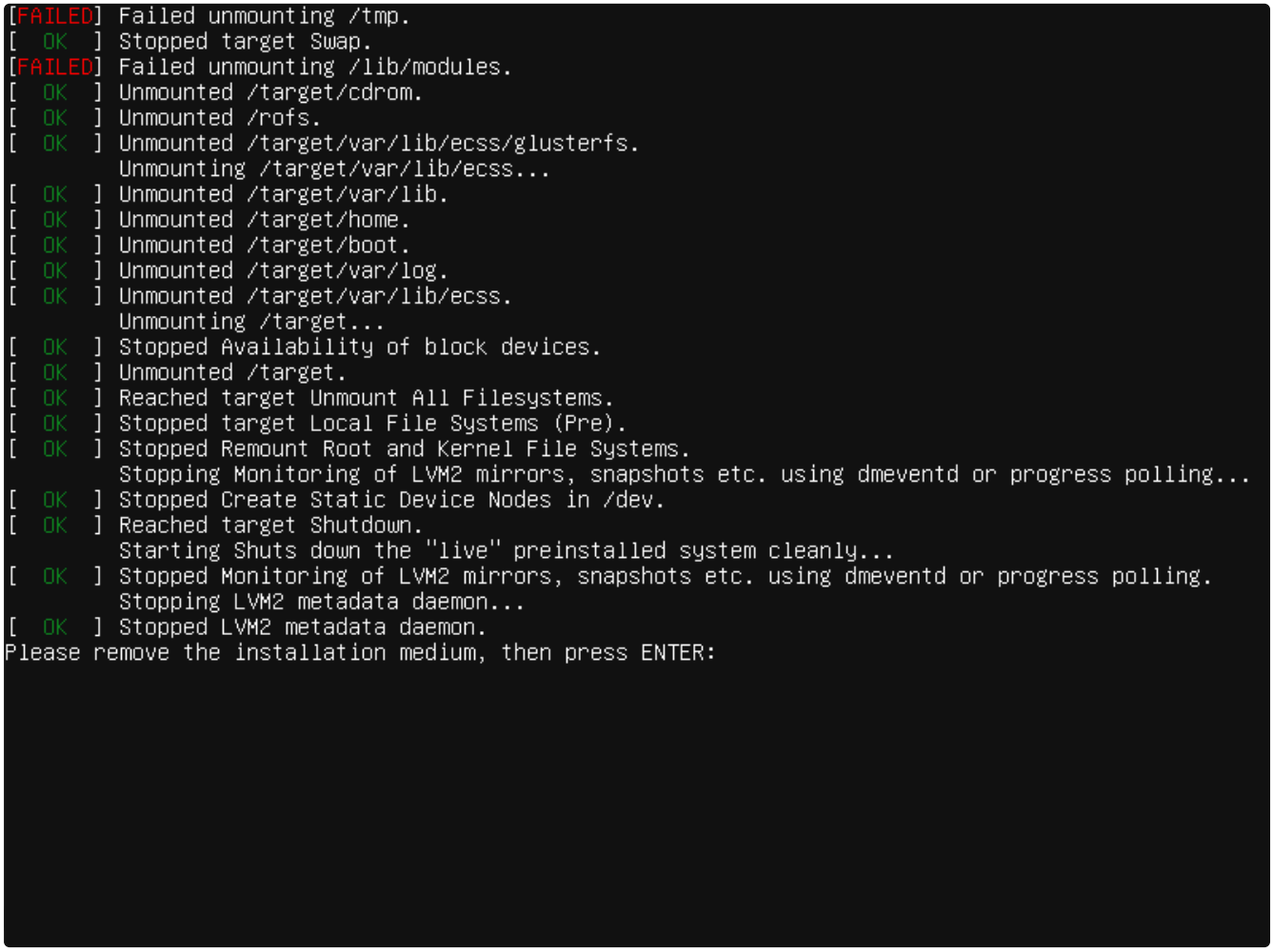
Подтвердить



Ждем появления сообщения - Перезагрузить сейчас



Извлекаем инсталляционный диск/флешку и Enter



Ubuntu Linux Server установлен

Таблица 1 — Вариант размещения информации в файловой системе на физических носителях для серверов

1	Загрузочный раздел операционной системы (создается автоматически)	boot	raid 1:hdd1,hdd2	boot	/boot	ext4	1 Gb	Первичный
2	Корневой раздел операционной системы	root	raid 1:hdd1,hdd2	root	/	ext4	30 Gb	Логический
3	Информация локальных баз данных	mnesia	raid 1:hdd1, hdd2	mnesia	/var/lib/ecss	ext4	10 Gb	Логический

4	Распределенная БД для хранения медиаресурсов	glusterfs	raid 1:hdd1, hdd2 или hdd3	glusterfs	/var/lib/ecss/glusterfs*	ext4	Max Gb	Логический
5	Журналы функционирования подсистем ОС	log	raid 1:hdd1,hdd2 или hdd3	log	/var/log	ext4	20 Gb	Логический
6	Журналы функционирования подсистем ECSS	ecss_log	raid 1:hdd1,hdd2 или hdd3	ecss_log	/var/log/ecss	ext4	20 Gb	Логический
7	Базы данных	ecss_db	raid 1:hdd1,hdd2 или hdd3	ecss_db	/srv/ecss/ecss-postgres-bdr-ssw/	ext4	100–400 Gb**	Логический
8	Файлы пользователя	home	raid 1:hdd1,hdd2 или hdd3	home	/home	ext4	10 Gb	Логический

⚠ * Если сервер не будет работать в кластере, то вместо **glusterfs** создается раздел **/var/lib/ecss/restfs**.
** Рекомендуемое значение для серий **Light, Light+, Midi** — 100 Gb. Рекомендуемое значение для серии **Heavy** — 200 Gb, **Super Heavy** — 400 Gb.

Для работы системы необходимо как минимум 256 Gb свободного пространства.

На серверах системы необходимо настроить параметр **"hostname".**

На всех серверах системы желательно указать одинаковое имя пользователя (любое, кроме **ssw**). Лицензия ECSS-10 привязывается к ключу eToken/ruToken и к имени компьютера (**hostname**). Системный пользователь **ssw** создается при инсталляции пакета **ecss-user**.

✓ При установке системы в кластере, **рекомендуемое** значение для первого сервера — **ecss1**, для второго — **ecss2**.

8.5 Настройка сети

Согласно параметрам, заданным в техническом задании (далее ТЗ), выполните сетевые настройки. В данном примере считается, что уже установлена необходимая операционная система.

Рекомендуется разделять трафик, используемый в разных целях. Например, трафик управления и трафик VoIP. Для этого создаются 2 или более vlan. При небольшой нагрузке для простоты можно обойтись и одним vlan, но это может в будущем доставлять неудобства при снятии дампов трафика и его анализе. В соответствии с техническим заданием на vlan настраиваются IP-адреса самого хоста, шлюзы, DNS и маршрутизация в другие сети.

В данном примере согласно ТЗ будем использовать адреса (в скобках — различия для ecss2):

- 10.0.10.51(52)/24 — для управления, vlan 10;
- 10.0.20.51(52)/24 — core, vlan 20;
- 10.0.20.31(32)/24 — виртуальные (vrrp) для VoIP;
- 10.0.10.1 — шлюз и dns для доступа во внешнюю сеть;
- 10.0.20.1 — шлюз в подсеть 10.0.3.0/24;

- 10.136.16.211, 10.136.16.212 — адреса NTP-серверов, доступ до которых осуществляется через шлюз 10.0.10.1.

Внутри серверной платформы существует адресная структура и используются внутренние адреса для взаимодействия между подсистемами (нодами) в кластере. Например, внутренний адрес для кластера на одном сервере 10.0.20.51, при этом ядро (ecss-core) взаимодействует с сервером обработки мультимедиа-данных (ecss-media-server). Их взаимодействие происходит с использованием одного и того же адреса, но у каждой программной части свой транспортный порт: ecss-core — 5000, ecss-msr — 5040.

Для всех нод кластера определяется единый адрес для обращения к базе данных, например, адрес 10.0.10.10. Таким образом, выполняется условие единообразия, при котором все ноды кластера имеют полностью одинаковые данные о текущем состоянии динамических составляющих программного коммутатора (например, история вызовов).

Сначала настраиваются сетевые интерфейсы.
В Ubuntu 22 для их настройки используется утилита **netplan**:

```
sudo nano /etc/netplan/ecss_netplan.yaml
```

В конфигурациях для каждого хоста в первую очередь мы объявляем секцию *ethernets,* в которой описываются существующие в системе ethernet-интерфейсы, используемые в дальнейшем. Для каждого интерфейса необходимо отключить использование динамического распределения адресов (DHCP).

Следующим разделом описываются агрегированные каналы — *bonds*. В зависимости от выбранного варианта включения в сеть настраивается режим резерва 1:1 (*active - backup*) или LACP (802.3ad).

Далее настраиваются vlan, на которых опционально определяются шлюзы для связи с внешним миром и адреса DNS-серверов, а также непосредственно IP-адреса для каждого интерфейса.

ВАЖНО

Обратите внимание, что во время редактирования netplan необходимо соблюдать правила разметки YAML:

- Обязательное наличие двух пробелов перед каждой строкой (кроме network);
- Каждая подсекция дополнительно сдвигается на 2 пробела:

→ Секция |network
→ Подсекция |_'_'bonds:
→ Подсекция описания секции bonds |_'_'_'_'bonded_one:
→ и т.д. |_'_'_'_'...

- Перед знаком ":" нет пробела, после — один пробел;
- Перед знаком "-" количество пробелов такое, как будто начинается новый подраздел, после — один пробел.

8.5.1 Пример netplan для режима active-backup

Netplan для интерфейсов сервера ecss1 (/etc/netplan/ecss_netplan.yaml)

```
# Netplan для хоста ecss1 программного коммутатора
# Обратите внимание на обязательное наличие минимум двух пробелов в каждой строке и секции (кроме строки секции network)

network:
  version: 2 # Версия netplan
  renderer: networkd # Исполнитель конфигураций netplan
  ethernets: # Раздел описания интерфейсов ethernet
    eth0: # Название интерфейса
      dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
    eth1:
      dhcp4: no
    eth2:
      dhcp4: no
    eth3:
      dhcp4: no

  bonds: # Раздел описания связанных (bonding) интерфейсов
    bond1: # Название bonding-интерфейса
      interfaces: # Секция определения связанного интерфейса
        - eth0
        - eth1
        - eth2
        - eth3
      parameters: # Раздел определения параметров связанного интерфейса
        primary-reselect-policy: failure # позволяет избежать лишних переключений
        gratuitous-arp: 5 # Способствует более быстрому переключению
        all-slaves-active: true # принимать входящие кадры на backup интерфейсах
        up-delay: 1000 # сделать
```

Netplan для интерфейсов сервера ecss2 (/etc/netplan/ecss_netplan.yaml)

```
# Netplan для хоста ecss2 программного коммутатора
# Обратите внимание на обязательное наличие минимум двух пробелов в каждой строке и секции (кроме строки секции network)

network:
  version: 2 # Версия netplan
  renderer: networkd # Исполнитель конфигураций netplan
  ethernets: # Раздел описания интерфейсов ethernet
    eth0: # Название интерфейса
      dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
    eth1:
      dhcp4: no
    eth2:
      dhcp4: no
    eth3:
      dhcp4: no

  bonds: # Раздел описания связанных (bonding) интерфейсов
    bond1: # Название bonding-интерфейса
      interfaces: # Секция определения связанного интерфейса
        - eth0
        - eth1
        - eth2
        - eth3
      parameters: # Раздел определения параметров связанного интерфейса
        primary-reselect-policy: failure # позволяет избежать лишних переключений
        gratuitous-arp: 5 # Способствует более быстрому переключению
        all-slaves-active: true # принимать входящие кадры на backup интерфейсах
        up-delay: 1000 # сделать задержку в одну секунду
        mode:  active-backup # Режим
```

Netplan для интерфейсов сервера ecss1 (/etc/netplan/ecss_netplan.yaml)

```
задержку в одну секунду
        mode:  active-backup # Режим
резервирования, один  активным, а
остальные в резерве
        mii-monitor-interval: 100 #
интервал мониторинга интерфейса(мс)
        primary: eth0 # Секция
определения основного интерфейса
      optional: false # Определение,
является ли интерфейс обязательным при
старте

  vlans:
    net.10: # Интерфейс управления
        id: 10
        link: bond1
        addresses: [10.0.10.51/24]
        gateway4: 10.0.10.1 # Адрес
шлюза
        nameservers:
          addresses: [10.0.10.1,
8.8.8.8] # Адреса серверов DNS
        routes: # Маршрутизация в
подсеть к NTP
          - to: 10.136.16.0/24
            via: 10.0.10.1 # Адрес
шлюза в данную подсеть
            on-link: true # Определяет,
что указанные маршруты непосредственно
связаны с интерфейсом
    net.20: # Интерфейс для VoIP
        id: 20
        link: bond1
        addresses: [10.0.20.51/24]
        routes:
          - to: 10.0.3.0/24
            via: 10.0.20.1
            on-link: true
```

Netplan для интерфейсов сервера ecss2 (/etc/netplan/ecss_netplan.yaml)

```
резервирования, один  активным, а
остальные в резерве
        mii-monitor-interval: 100 #
интервал мониторинга интерфейса(мс)
        primary: eth0 # Секция
определения основного интерфейса
      optional: false # Определение,
является ли интерфейс обязательным при
старте

  vlans:
    net.10: # Интерфейс управления
        id: 10
        link: bond1
        addresses: [10.0.10.52/24]
        gateway4: 10.0.10.1 # Адрес
шлюза
        nameservers:
          addresses: [10.0.10.1,
8.8.8.8] # Адреса серверов DNS
        routes: # Маршрутизация в
подсеть к NTP
          - to: 10.136.16.0/24
            via: 10.0.10.1 # Адрес
шлюза в данную подсеть
            on-link: true # Определяет,
что указанные маршруты непосредственно
связаны с интерфейсом
    net.20: # Интерфейс для VoIP
        id: 20
        link: bond1
        addresses: [10.0.20.52/24]
        routes:
          - to: 10.0.3.0/24
            via: 10.0.20.1
            on-link: true
```

Следующие настройки **bonds** обязательны для сервера ECSS, чтобы схема работала верно:

mode: active-backup - задаёт режим работы, когда один из линков выбирается активным, а остальные остаются в резерве;
primary-reselect-policy: failure - указывает, что выбирать новый активный линк следует только тогда, когда текущий активный линк переходит в состоянии аварии. Это позволяет избежать лишних переключений;
gratuitous-arp: 5 - при смене активного линка в сторону коммутатора отправляются пять запросов gratuitous ARP, чтобы обновить на нём таблицу коммутации. Способствует более быстрому переключению;

all-slaves-active: true - заставляет принимать входящие кадры на backup интерфейсах. Таким образом балансировка трафика на MESe не мешает работе. Данные в сторону сервера идут со всех линков, а сервер отправляет данные только с active линка;
mii-monitor-interval: 100 - активирует мониторинг линков через интерфейс MII и указывает интервал опроса в 100мс;
up-delay: 1000 - указывает считать поднявшийся интерфейс доступным для работы не сразу, а сделать задержку в одну секунду после того, как интерфейс поднялся. Необходимо для того, чтобы избежать лишних переключений в случае, когда порт "прыгает" несколько раз из состояния "включено" в состояние "выключено" и обратно.

8.5.2 Пример netplan для режима 802.3ad

Netplan для интерфейсов сервера ecss1 (/etc/netplan/ecss_netplan.yaml)	**Netplan для интерфейсов сервера ecss2 (/etc/netplan/ecss_netplan.yaml)**

```
# Netplan для хоста ecss1 программного коммутатора
# Обратите внимание на обязательное наличие минимум двух пробелов в каждой строке и секции (кроме строки секции network)

network:
  version: 2 # Версия netplan
  renderer: networkd # Исполнитель конфигураций netplan
  ethernets: # Раздел описания интерфейсов ethernet
    eth0: # Название интерфейса
      dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
    eth1:
      dhcp4: no
    eth2:
      dhcp4: no
    eth3:
      dhcp4: no

  bonds: # Раздел описания связанных (bonding) интерфейсов
    bond1: # Название bonding-интерфейса
      interfaces: # Секция определения связанного интерфейса
        - eth0
        - eth1
        - eth2
        - eth3
      parameters: # Раздел определения параметров связанного интерфейса
        mode: 802.3ad # Режим LACP
```

```
# Netplan для хоста ecss2 программного коммутатора
# Обратите внимание на обязательное наличие минимум двух пробелов в каждой строке и секции (кроме строки секции network)

network:
  version: 2 # Версия netplan
  renderer: networkd # Исполнитель конфигураций netplan
  ethernets: # Раздел описания интерфейсов ethernet
    eth0: # Название интерфейса
      dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
    eth1:
      dhcp4: no
    eth2:
      dhcp4: no
    eth3:
      dhcp4: no

  bonds: # Раздел описания связанных (bonding) интерфейсов
    bond1: # Название bonding-интерфейса
      interfaces: # Секция определения связанного интерфейса
        - eth0
        - eth1
        - eth2
        - eth3
      parameters: # Раздел определения параметров связанного интерфейса
        mode: 802.3ad # Режим LACP
        mii-monitor-interval: 100 #
```

Netplan для интерфейсов сервера ecss1 (/etc/netplan/ecss_netplan.yaml)

```
        mii-monitor-interval: 100 #
интервал мониторинга интерфейса(мс)
        primary: eth0 # Секция
определения основного интерфейса
      optional: false # Определение,
является ли интерфейс обязательным при
старте

  vlans:
    net.10: # Интерфейс управления
        id: 10
        link: bond1
        addresses: [10.0.10.51/24]
        gateway4: 10.0.10.1 # Адрес
шлюза
        nameservers:
          addresses: [10.0.10.1,
8.8.8.8] # Адреса серверов DNS
        routes: # Маршрутизация в
подсеть к NTP
          - to: 10.136.16.0/24
            via: 10.0.10.1 # Адрес
шлюза в данную подсеть
            on-link: true # Определяет,
что указанные маршруты непосредственно
связаны с интерфейсом
    net.20: # Интерфейс для VoIP
        id: 20
        link: bond1
        addresses: [10.0.20.51/24]
        routes:
          - to: 10.0.3.0/24
            via: 10.0.20.1
            on-link: true
```

Netplan для интерфейсов сервера ecss2 (/etc/netplan/ecss_netplan.yaml)

```
интервал мониторинга интерфейса(мс)
        primary: eth0 # Секция
определения основного интерфейса
      optional: false # Определение,
является ли интерфейс обязательным при
старте

  vlans:
    net.10: # Интерфейс управления
        id: 10
        link: bond1
        addresses: [10.0.10.52/24]
        gateway4: 10.0.10.1 # Адрес
шлюза
        nameservers:
          addresses: [10.0.10.1,
8.8.8.8] # Адреса серверов DNS
        routes: # Маршрутизация в
подсеть к NTP
          - to: 10.136.16.0/24
            via: 10.0.10.1 # Адрес
шлюза в данную подсеть
            on-link: true # Определяет,
что указанные маршруты непосредственно
связаны с интерфейсом
    net.20: # Интерфейс для VoIP
        id: 20
        link: bond1
        addresses: [10.0.20.52/24]
        routes:
          - to: 10.0.3.0/24
            via: 10.0.20.1
            on-link: true
```

Так же рекомендуется проверить отсутствие в каталоге **/etc/netplan/** еще каких либо файлов, если другие файлы присутствуют , то их нужно переместить в другой каталог или удалить, в противном случае возможна некорректная настройка сетевых интерфейсов и некорректная работа SSW.

Применим установленные параметры командой:

на обоих хостах

```
sudo netplan apply
```

Просмотреть получившиеся настройки можно при помощи команд **ifconfig** или **ip a.**

На серверах системы необходимо настроить параметр **"hostname".**

На всех серверах системы желательно указать одинаковое имя пользователя (любое, кроме **ssw**). Лицензия ECSS-10 привязывается к ключу eToken/ruToken и к имени компьютера (**hostname**), поэтому необходимо использовать стандартные значения. Системный пользователь **ssw** создается при инсталляции пакета **ecss-user**.

При установке системы в кластере значение для первого сервера — **ecss1**, для второго — **ecss2**. Другие имена хостов возможны **только** при согласовании проекта, это потребуется для генерации лицензий.

Указать имя хоста: **ecss1 / ecss2** в файле /etc/hostname:

на обоих хостах

```
sudo nano /etc/hostname
```

8.5.3 Настройка /etc/hosts

После настройки *netplan* необходимо указать, что внутренний адрес *10.0.10.X* принадлежит соответствующему серверу *ecssX*. Для этого настраиваем */etc/hosts*:

Настройка hosts для ecss1 (/etc/hosts)

```
127.0.0.1   localhost # Адрес локальной
петли, используется некоторыми
сервисами ecss
10.0.10.51  ecss1 # Адрес хоста
10.0.10.52  ecss2 # Внутренний адрес
другого хоста
```

Настройка hosts для ecss2 (/etc/hosts)

```
127.0.0.1 localhost # Адрес локальной
петли, используется некоторыми сервисами
ecss
10.0.10.52 ecss2 # Адрес хоста
10.0.10.51 ecss1 # Внутренний адрес
другого сервера
```

Теперь, если вызвать утилиту ping на *ecssX,* можно обратиться к соседнему серверу.

Обращение к ecss2 с ecss1

```
ping ecss2
PING ecss2 (10.0.10.52) 56(84) bytes of
data.
64 bytes from ecss2 (10.0.10.52):
icmp_seq=1 ttl=64 time=0.047 ms
```

Обращение к ecss1 с ecss2

```
ping ecss1
PING ecss1 (10.0.10.51) 56(84) bytes of
data.
64 bytes from ecss1 (10.0.10.51):
icmp_seq=1 ttl=64 time=0.032 ms
```

Далее следует организовать доступ между серверами через ssh по rsa-ключам - **без использования пароля**.

Генерируем rsa-ключ следующий командой (выполняем команду **без sudo**, чтобы ключ сгенерировался для текущего пользователя), при выполнении команды будут заданы три вопроса, можно использовать значение по умолчанию, нажав три раза Enter:

на обоих хостах

```
ssh-keygen
```

на ecss1

```
ssh-copy-id ecss2
```

Таким же образом на хосте *ecss2*.

на ecss2

```
ssh-copy-id ecss1
```

проврете подключение между серверами, со стороны сервера ecss1 - **ssh ecss2** соединение должно быть установлено **без запроса пароля** . Аналогичным образом со стороны сервера ecss2 - **ssh ecss1**

8.6 "Оптимизация" Операционной Системы

8.6.1 Выставить параметры ОС в режим производительности

Используем утилиту cpufrequtils.

на обоих хостах

```
sudo apt install -y cpufrequtils
```

по умолчанию после инсталляции Ubuntu использует режим "ondemand" - "по запросу" (производительность CPU по запросу приложений, экономит электроэнергию , но ниже производительность):

на обоих хостах

```
cat /etc/init.d/cpufrequtils | grep GOVERNOR=
```

в выходном сообщении системы , режим работы по умолчанию после инсталляции - "ondemand" :

```
GOVERNOR="ondemand"
```

Установить режим - результативность/производительность - в файле /etc/init.d/cpufrequtils значение "**ondemand**" заменить на "**performance**"

на обоих хостах

```
sudo sed -i 's/GOVERNOR="ondemand"/GOVERNOR="performance"/g' /etc/init.d/cpufrequtils
```

Перезапустить утилиту:

на обоих хостах

```
sudo /etc/init.d/cpufrequtils restart
```

Затем выполнить команду:

на обоих хостах

```
sudo systemctl daemon-reload
```

8.6.2 Отключить SWAP

Ubuntu сервер SSW работает **в реальном масштабе времени** ,
поэтому **все** необходимые **данные должны находится в оперативной памяти,** использование файла подкачки (swap-файл - /swap. img) может привести к увеличению времени обработки вызовов приложения SSW ECSS10 , **что недопустимо**. Swap - отключаем.
Выполнить последовательно три команды:

Отключение Swap:

на обоих хостах

```
sudo swapoff -a
```

Удалить файл swap.img.

на обоих хостах

```
sudo rm /swap.img
```

Закомментировать строку - /swap.img none swap sw 0 0 - в файле /etc/fstab -

привести ее к виду # /swap.img none swap sw 0 0

либо **удалить** эту строчку (/swap.img none swap sw 0 0)

на обоих хостах

```
sudo nano /etc/fstab
```

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/731728e2-4d6b-499a-afea-9362fd6726b2 / ext4 defaults 0 1
# /swap.img none swap sw 0 0
```

Для проверки выполните команду **free -h:**

на обоих хостах

```
free -h
```

Размер Swap равен 0 - то есть он выключен

```
free -h
               total        used        free      shared  buff/cache   available
Mem:           3,9G        110M        3,2G        820K        535M        3,5G
Swap:            0B          0B          0B
```

8.6.3 Установка часового пояса

При инсталляции Ubuntu-22 **не предлагается** установить часовой пояс(по умолчанию устанавливается время UTC). Его нужно установить **вручную** (для корректной работы системы тарификации , работ по расписанию и т.д.), например:

на обоих хостах

```
sudo timedatectl set-timezone Asia/Novosibirsk
```

В случае выноса компонентов системы на разные серверы/ВМ (ecss-node/ecss-msr) необходимо указывать единый часовой пояс, для всех компонентов системы.

8.6.4 Улучшение работы высоконагруженных серверов

Улучшить работу высоконагруженных серверов можно увеличив лимит открытых файлов.

Для установки лимита открытых файлов необходимо:

Проверить текущий лимит командой:

на обоих хостах

```
ulimit -a
```

результат :

```
eltex@ecss1:~$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
scheduling priority             (-e) 0
file size               (blocks, -f) unlimited
pending signals                 (-i) 15515
max locked memory       (kbytes, -l) 65536
max memory size         (kbytes, -m) unlimited
open files                      (-n) 1024
pipe size            (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
real-time priority              (-r) 0
stack size              (kbytes, -s) 8192
cpu time               (seconds, -t) unlimited
max user processes              (-u) 15515
virtual memory          (kbytes, -v) unlimited
file locks                      (-x) unlimited
```

Данного лимита (open files **1024**) **недостаточно** для нормальной работы высоконагруженных серверов.

Установить лимит открытых файл для каждого пользователя:

на обоих хостах

```
sudo sed -i  '55i\*               soft    nproc           65536\n*               hard
nproc           131072\n*               soft    nofile          65536\n*               hard
nofile          131072\nroot            -       memlock         unlimited' /etc/security/
limits.conf
```

Результат:

```
# /etc/security/limits.conf
#
#Each line describes a limit for a user in the form:
#
#<domain>        <type>  <item>  <value>
#
#Where:
#<domain> can be:
#        - a user name
#        - a group name, with @group syntax
```

```
#        - the wildcard *, for default entry
#        - the wildcard %, can be also used with %group syntax,
#                 for maxlogin limit
#        - NOTE: group and wildcard limits are not applied to root.
#          To apply a limit to the root user, <domain> must be
#          the literal username root.
#
#<type> can have the two values:
#        - "soft" for enforcing the soft limits
#        - "hard" for enforcing hard limits
#
#<item> can be one of the following:
#        - core - limits the core file size (KB)
#        - data - max data size (KB)
#        - fsize - maximum filesize (KB)
#        - memlock - max locked-in-memory address space (KB)
#        - nofile - max number of open files
#        - rss - max resident set size (KB)
#        - stack - max stack size (KB)
#        - cpu - max CPU time (MIN)
#        - nproc - max number of processes
#        - as - address space limit (KB)
#        - maxlogins - max number of logins for this user
#        - maxsyslogins - max number of logins on the system
#        - priority - the priority to run user process with
#        - locks - max number of file locks the user can hold
#        - sigpending - max number of pending signals
#        - msgqueue - max memory used by POSIX message queues (bytes)
#        - nice - max nice priority allowed to raise to values: [-20, 19]
#        - rtprio - max realtime priority
#        - chroot - change root to directory (Debian-specific)
#
#<domain>      <type>  <item>         <value>
#

#*               soft    core            0
#root            hard    core            100000
#*               hard    rss             10000
#@student        hard    nproc           20
```

```
#@faculty       soft    nproc           20
#@faculty       hard    nproc           50
#ftp            hard    nproc           0
#ftp            -       chroot          /ftp
#@student       -       maxlogins       4
*               soft    nproc           65536
*               hard    nproc           131072
*               soft    nofile          65536
*               hard    nofile          131072
root            -       memlock         unlimited
# End of file
```

> **ВАЖНО**
>
> Установку пакетов требуется делать НЕ из-под пользователя **ssw**.

8.7 Обновление программного обеспечения операционной системы

Для установки системы ECSS-10 необходимо добавить репозиторий ELTEX.

на обоих хостах

```
sudo sh -c "echo 'deb [arch=amd64] http://archive.eltex.org/ssw/jammy/3.18 stable main extras external' > /etc/apt/sources.list.d/eltex-ecss10-stable.list"
```

Далее необходимо выполнить импорт ключа командой:

на обоих хостах

```
sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 33CB2B750F8BB6A5
```

Для обновления ОС необходимо выполнить команды:

на обоих хостах

```
sudo apt update
```

⚠ Если вы видите такое сообщение системы:

```
W: http://archive.eltex.org/ssw/jammy/3.18/dists/unstable/InRelease: Key is
stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the
DEPRECATION section in apt-key(8) for details.
```

Выполните следующую команду:

на обоих хостах

```
sudo cp /etc/apt/trusted.gpg /etc/apt/trusted.gpg.d
```

на обоих хостах

```
sudo apt upgrade
```

8.8 Установка и настройка программного обеспечения

установка пакетов deb

```
   ┌ версия ПО ECSS. Сейчас в заявке на сертификацию она заявлена как 3.

   │ Мы её менять не будем до следующей сертификации или
  │ выпуска принципиально новой версии системы.

   │   ┌ версия System Release. Общая версия для всех компонент входящих в конкретный релиз.

   │      │ Меняем её централизовано,
   │   │ принимаем решение о выпуске нового релиза. Релизы между собой обычно не совместимы.
  ┌┴┐ ┌──┴──┐
ECSS.SysRel.SubMaj.SubMin
              └──┬──┘ └──┬──┘
        │    └ версия Subsystem minor. Минорную версию подсистемы устанавливает разработчик
подсистемы.

                  │          Минорную версию
        │     меняем при добавлении патча. Как правило, минорные версии в рамках одной мажорной
версии
        │     совместимы между собой и отличаются в рамках конкретных патчей.

                 └ версия Subsystem major. Мажорная версия устанавливается разработчиком
подсистемы.

                    Мажорную версию подсистемы необходимо менять при внесении в подсистему
существенных изменений.
```

Утилита установки пакетов APT анализирует версию пакета слева направо, то есть мы имеем пакет 14.14.7.7 ,
в репозитории находятся пакеты
14.14.7.8
14.14.7.9
14.14.8.1
14.14.20 -14.14.28
то при выполнении команды **sudo apt install имя пакета** → будет автоматически проверен и установлен пакет 14.14.28 , так как он самый последний, анализ будет выполнен по 3-й позиции (major), анализ по 4-й (minor) выполнен не будет (аналогичным образом будет выполнятся команда **sudo apt upgrade**).
В случае если для конкретной ситуации требуется перейти с версии 14.14.7.7 на версию 14.14.7.9 , стандартная команда → **sudo apt upgrade** , нам не поможет, так как будет выбран самый новый пакет, в данной ситуации нам **необходимо в явном виде** указать какую версию пакета мы хотим установить, в данном примере мы **должны** выбрать команду → **sudo apt install имя пакета=14.14.7.9** . Обычно это необходимо для тестирования определенного патча, для стандартных обновлений достаточно выбора привычной команды установки / обновления пакета

Установите все предложенные пакеты:

на обоих хостах

```
sudo apt install -y ntp ntpdate tcpdump vlan dnsmasq aptitude atop ethtool htop iotop mc
minicom mtr-tiny nmap pptpd pv screen ssh tftpd vim sngrep tshark cpanminus gnuplot libgraph-
easy-perl debconf-utils
```

Также установите пакеты для системы с резервированием:

на обоих хостах

```
sudo apt install -y ifenslave keepalived attr
```

Пакеты ПО которые рекомендуется установить для работы

Список **обязательного** сервисного программного обеспечения:

```
sudo apt install -y ntp tcpdump vlan dnsmasq
```

ntp	NTP-сервер
tcpdump	сниффер пакетов
vlan	управление VLAN
dnsmasq	легковесный DNS/DHCP-сервер

Список **рекомендуемого** диагностического и вспомогательного программного обеспечения:

```
sudo apt install -y aptitude atop ethtool htop mc screen ssh tftpd sngrep tshark gnuplot
libgraph-easy-perl debconf-utils iotop ncdu
```

aptitude	установка программ из репозиториев, рекомендуется использовать вместо программы apt/apt-get
atop	мониторинг загрузки хоста с функцией периодического сохранения информации в файлы
ethtool	просмотр статистики сетевых интерфейсов
htop	мониторинг процессов
mc	файловый менеджер
screen	мультиплексор терминалов
ssh	сервер и клиент SSH
tftpd	TFTP-сервер
sngrep	трассировка sip

tshark	консольный аналог wireshark
gnuplot	вывод графиков статистики
libgraph-easy-perl	Perl-модуль для преобразования или рендеринга графиков (в ASCII, HTML, SVG или через Graphviz)
debconf-utils	набор утилит для работы с базой debconf
iotop	инструмент для мониторинга использования ввода-вывода (IO) на диске в реальном времени в Linux
ncdu	утилита для поиска больших директорий в системе Linux

⚠ Данное программное обеспечение не требуется для работы системы ECSS-10, однако может упростить сервисное обслуживание системы и её отдельных компонентов со стороны инженеров эксплуатации и техподдержки.

Список **обязательных** пакетов для схем **с резервированием**:

```
sudo apt install -y ifenslave-2.6 keepalived attr
```

ifenslave-2.6	управление BOND-интерфейсами
keepalived	сервис мониторинга серверов/служб в кластере
attr	сервис управления атрибутами файловых систем

Список **дополнительных** пакетов для схем **с резервированием**:

```
sudo apt install -y bridge-utils ethtool
```

bridge-utils	управление bridge-интерфейсами
ethtool	управление и мониторинг сетевых интерфейсов

⚠ Перед началом установки пакетов ecss , нужно убедится в соответствии полосы пропускания СПД необходимым требованиям.
Для этого выполнить команду **sudo ethtool <имя интерфейса>** для всех физических интерфейсов.
Пример:

```
sudo ethtool net.20
Settings for net.20:
    Supported ports: [ TP ]
    Supported link modes:   10baseT/Half 10baseT/Full
                            100baseT/Half 100baseT/Full
                            1000baseT/Full
    Supported pause frame use: No
    Supports auto-negotiation: Yes
    Supported FEC modes: Not reported
    Advertised link modes:  10baseT/Half 10baseT/Full
                            100baseT/Half 100baseT/Full
                            1000baseT/Full
    Advertised pause frame use: No
    Advertised auto-negotiation: Yes
    Advertised FEC modes: Not reported
    Speed: 1000Mb/s
    Duplex: Full
    Auto-negotiation: on
    Port: Twisted Pair
    PHYAD: 0
    Transceiver: internal
    MDI-X: off (auto)
    Link detected: yes
```

Проверить значение следующих параметров:
Advertised auto-negotiation: **Yes**
Speed: **1000Mb/s** (не менее)
Duplex: **Full**

8.9 Инсталляция обязательных пакетов ECSS

8.9.1 Установка пакета ecss-dns-env

Выполнить следующую команду:

на обоих хостах

```
sudo apt install -y ecss-dns-env
```

Настройщик предложит выбрать разделы для настройки по вопросам ниже. **Нужно выбрать broker.** (для примера 10.0.10.51 и 10.0.10.52)

Вопрос ecss-dns-env	Ответы для ecss1	Ответы для ecss2
[Primary broker] Введите адрес ([Primary broker] Enter ip)	**10.0.10.51** (необходимо ввести)	**10.0.10.51** (необходимо ввести)
[Secondary broker] Введите адрес ([Secondary broker] Enter ip)	**10.0.10.52** (необходимо ввести)	**10.0.10.52** (необходимо ввести)

ВАЖНО

В случае последующей коррекции IP-адресов следует использовать команду:

```
sudo dpkg-reconfigure ecss-dns-env
```

8.9.2 Установка Postgres

Примечание

Перед установкой `ecss-postgres-bdr-ssw` в системе, в `apt sources,` должен быть добавлен `docker` для этого выполнить следующие команды:

на обоих хостах

```
sudo install -m 0755 -d /etc/apt/keyrings
```

на всех хостах

```
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/
docker.asc
```

на всех хостах

```
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

на всех хостах

```
echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]
https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

на всех хостах - Выполнить обновление:

```
sudo apt update && sudo apt upgrade
```

8.9.3 Установка пакета ecss-postgres-bdr-ssw

Выполняется последовательно, первым устанавливается на сервере который работает в режиме "Master"

ecss1

```
sudo apt install -y ecss-postgres-bdr-ssw
```

При конфигурации пакета **ecss-postgres-bdr-ssw** будут заданы вопросы:

Вопросы ecss-postgres-bdr-ssw	Ответы для ecss1 (Master)
Текущий IP-адрес сетевого интерфейса.	**Выбрать** из списка IP-адрес для работы с Postgres
Установка в кластере?	Yes (значение по умолчанию)

Вопросы ecss-postgres-bdr-ssw	Ответы для ecss1 (Master)
Является ли этот узел мастером?	**Yes** (необходимо выбрать)

ecss2

```
sudo apt install -y ecss-postgres-bdr-ssw
```

При конфигурации пакета **ecss-postgres-bdr-ssw** будут заданы вопросы:

Вопросы ecss-postgres-bdr-ssw	Ответы для ecss2 (Slave)
Текущий IP-адрес сетевого интерфейса.	**Выбрать** из списка IP-адрес для работы с Postgres
Установка в кластере?	Yes (значение по умолчанию)
Является ли этот узел мастером?	No (значение по умолчанию)
IP-адрес, используемый для подключения к кластеру:	**Указать** IP-адрес Master (ecss1)

8.9.4 Проверить корректность настройки репликации

Во время установки в директорию `/srv/ecss/ecss-postgres-bdr-ssw` устанавливаются скрипт **postgresbdr_ssw.sh**

Для проверки репликации выполнить следующие команды:

Для проверки репликации выполнить команду на обоих хостах:

```
cd /srv/ecss/ecss-postgres-bdr-ssw
```

Для проверки репликации выполнить команду на обоих хостах:

```
sudo ./postgresbdr_ssw.sh check
```

Вывод на хостах **должен быль одинаковый:**

ecss1

```
sudo ./postgresbdr_ssw.sh check
Enter master ip please
Like 10.150.150.15
10.0.10.51
Enter slave ip please
10.0.10.52
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
таблица нод на слейве по ecss_storekeeper_db
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+----------------------------------------------------------------------------
----------
 7569499877813944343 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.52 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 rows)

таблица коннекций на слейве по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+----------------------------------------------------------------------------
----------
 7569499877813944343 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.52 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 rows)
```

ecss2

```
sudo ./postgresbdr_ssw.sh check
Enter master ip please
Like 10.150.150.15
10.0.10.51
Enter slave ip please
10.0.10.52
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
таблица нод на слейве по ecss_storekeeper_db
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+------------------------------------------------------------------------------
----------
 7569499877813944343 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.52 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 rows)
таблица коннекций на слейве по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+------------------------------------------------------------------------------
----------
 7569499877813944343 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.52 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 rows)
```

пример с ошибками

нет доступа до slave

```
sudo ./postgresbdr_ssw.sh check
[sudo] password for abf:
Sorry, try again.
[sudo] password for abf:
Enter master ip please
Like 10.150.150.15
10.0.10.51
Enter slave ip please
10.0.10.52
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
     node_sysid      |                                    node_local_dsn
                                  | node_init_from_dsn
---------------------
+-------------------------------------------------------------------------
-------------+--------------------
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1 |
(1 row)

таблица нод на слейве по ecss_storekeeper_db
psql: error: connection to server at "10.0.10.52", port 5439 failed:
Connection refused
    Is the server running on that host and accepting TCP/IP connections?
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                      conn_dsn
                                                                    |
---------------------
+-------------------------------------------------------------------------
-------------
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(1 row)
```

```
таблица коннекций на слейве по ecss_storekeeper_db
psql: error: connection to server at "10.0.10.52", port 5439 failed:
Connection refused
    Is the server running on that host and accepting TCP/IP connections?
```

ошибка несоответствия коннекций

```
sudo ./postgresbdr_ssw.sh check
Enter master ip please
Like 10.150.150.15
10.0.10.51
Enter slave ip please
10.0.10.52
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
     node_sysid      |                                        node_local_dsn
                                    | node_init_from_dsn
---------------------
+-------------------------------------------------------------------------
-------------+--------------------
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1 |
(1 row)
таблица нод на слейве по ecss_storekeeper_db
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                          conn_dsn
                                                                    |
---------------------
+-------------------------------------------------------------------------
-------------
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(1 row)
```

```
таблица коннекций на слейве по ecss_storekeeper_db
     conn_sysid      |                                    conn_dsn

---------------------
+---------------------------------------------------------------------------
-------------
 7569499877813944343 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.52 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(3 rows)
```

8.9.5 Установка ecss-node

Установка обязательного пакета **ecss-node** включает в себя установку и первоначальную настройку основных подсистем.

⚠ В системе уже должен быть установлен пакет **ecss-postgres-bdr-ssw**

Для установки пакета **ecss-node** выполните команду:

первоначально на ecss1 по окончанию установки пакета на ecss2

```
sudo apt install -y ecss-node
```

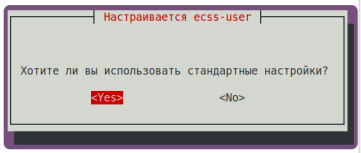
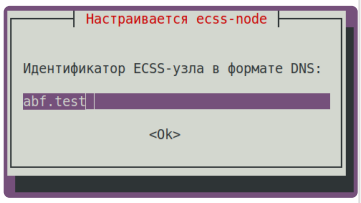
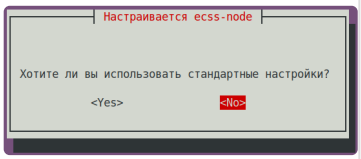
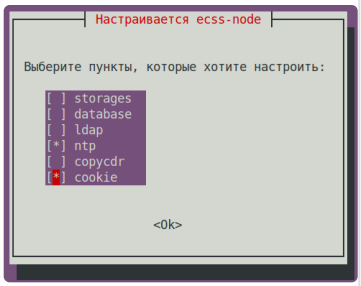
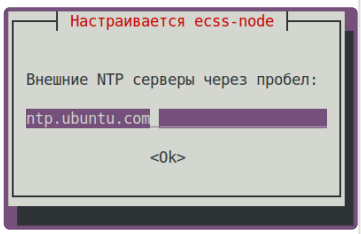
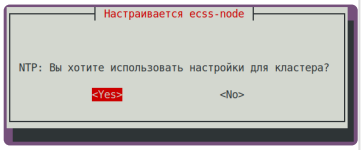
Во время установки пакета создается пользователь **ssw**, от имени которого запускаются все сервисы ecss*. Создаются необходимые каталоги, выполняется настройка DNS, идет настройка SSL-сертификатов.

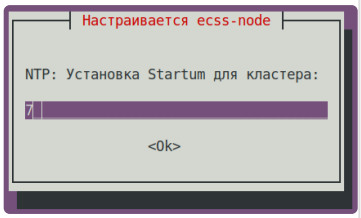
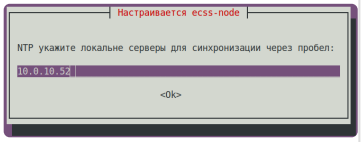
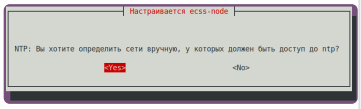
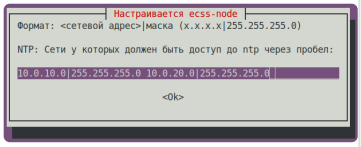
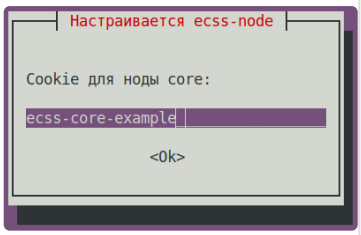
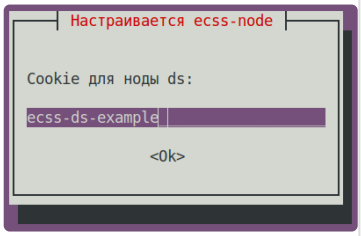
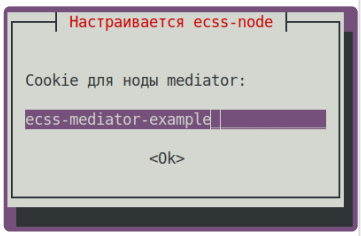
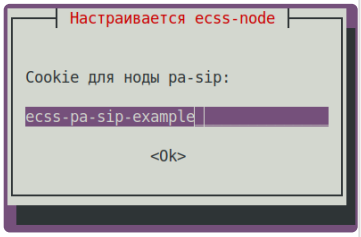
✔ При установки так же будет установлен пакет **ecss-user**

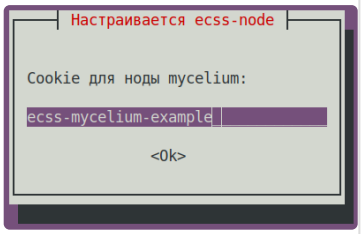
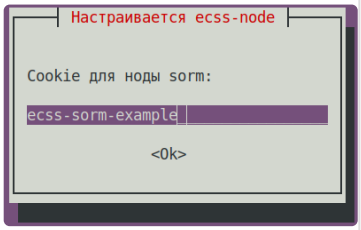
Установка на сервере ecss1

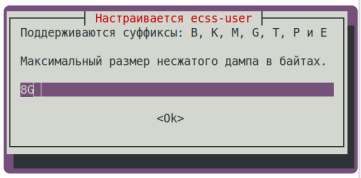
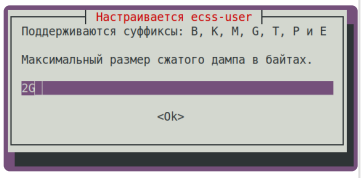
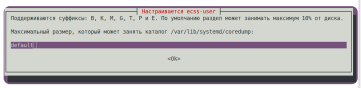
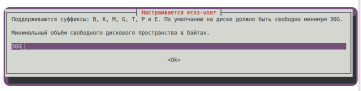
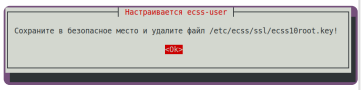
Во время установки будет предложено настроить параметры, необходимые для формирования конфигурационных файлов, примеры ответов следует ниже.

Вопросы ecss-user	Ответы для ecss1	Пример

Хотите ли вы использовать стандартные настройки?	Yes (значение по умолчанию)	
Вопросы ecss-node	**Ответы для ecss1**	**Пример**
Идентификатор ECSS-узла в формате DNS (Настройка имени кластера): (любое уникальное имя, в примере abf.test)	**abf.test** (необходимо ввести) Формат **ECSS_ID** Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире. `[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]`	
Хотите ли вы использовать стандартные настройки?	**No** (необходимо выбрать)	
Выберите пункты, которые хотите настроить:	**ntp + cookie** (необходимо выбрать)	
Внешние NTP серверы через пробел:	ntp.ubuntu.com (значение по умолчанию)	
NTP: Вы хотите использовать настройки для кластера?	Yes (значение по умолчанию)	

NTP: Установка Startum для кластера:	7 (значение по умолчанию)	
NTP укажите локальне серверы для синхронизации через пробел: (в примере ecss2 - 10.0.10.52)	**10.0.10.52** (необходимо ввести)	
NTP: Вы хотите определить сети вручную, у которых должен быть доступ до ntp?	Yes (значение по умолчанию)	
NTP: Сети у которых должен быть доступ до ntp через пробел: (в примере 10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0)	**10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0** (необходимо ввести)	
Cookie для ноды core: (указать уникальный cookie для core , в примере ecss-core-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-core-example** (необходимо ввести)	
Cookie для ноды ds: (указать уникальный cookie для ds , в примере ecss-ds-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-ds-example** (необходимо ввести)	
Cookie для ноды mediator: (указать уникальный cookie для mediator , в примере ecss-mediator-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mediator-example** (необходимо ввести)	
Cookie для ноды pa-sip: (указать уникальный cookie для pa-sip , в примере ecss-pa-sip-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-pa-sip-example** (необходимо ввести)	

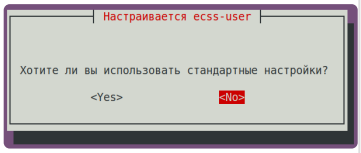
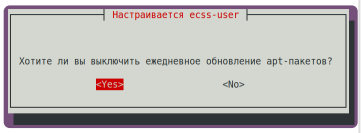
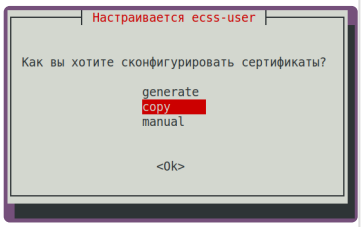
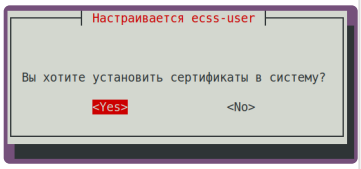
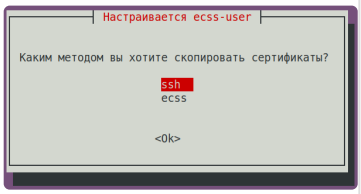
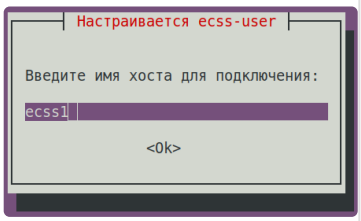
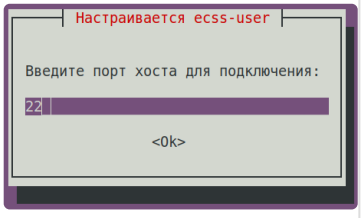
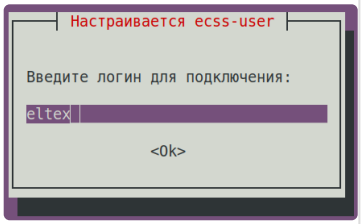
Cookie для ноды mycelium: (указать уникальный cookie для mycelium , в примере ecss-mycelium-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mycelium-example** (необходимо ввести)	
Cookie для ноды sorm: (указать уникальный cookie для sorm , в примере ecss-sorm-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-sorm-example** (необходимо ввести)	

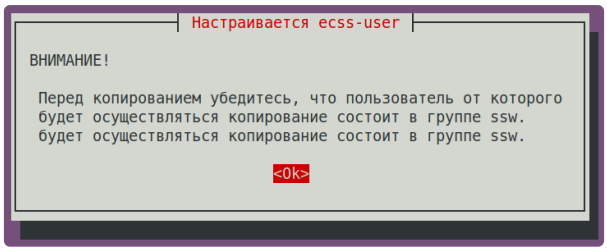
Вопросы ecss-user	Ответы для ecss1	Пример
Максимальный размер несжатого дампа в байтах.	8G (значение по умолчанию)	
Максимальный размер сжатого дампа в байтах.	2G (значение по умолчанию)	
Максимальный размер, который может занять каталог /var/lib/systemd/coredump:	default (значение по умолчанию)	
Минимальный объем свободного дискового пространства в байтах.	30G (значение по умолчанию)	
Сохраните в безопасное место и удалите файл /etc/ecss/ssl/ecss10root.key!	Ok (значение по умолчанию)	

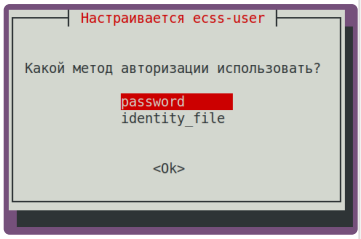
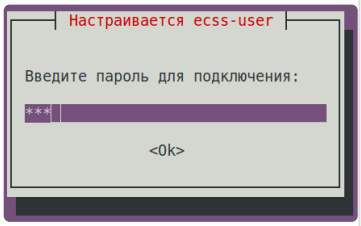
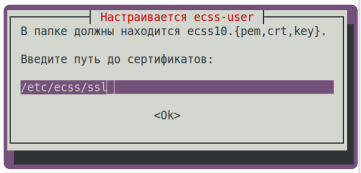
После установки пакета ecss-node на хосте ecss1, выполните команду **"sudo usermod -a -G ssw <Имя_пользователя>" на хосте ecss1**.
Затем **выполите установку на хосте ecss2.**
Используйте режим "Скопировать существующие сертификаты (copy) по ssh" с сервера ecss1

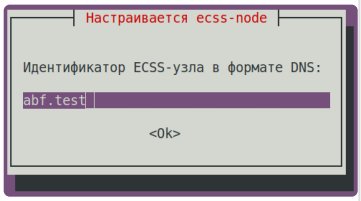
Установка на сервере ecss2

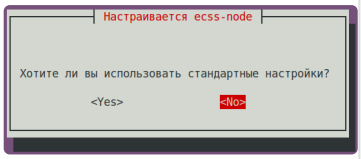
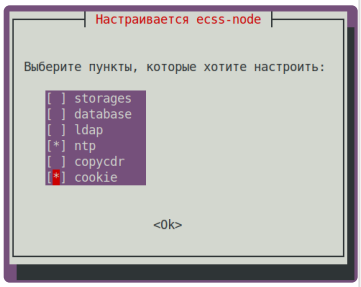
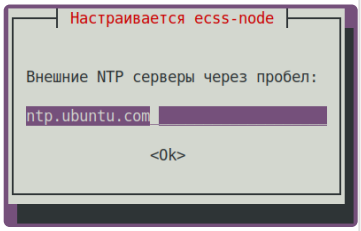
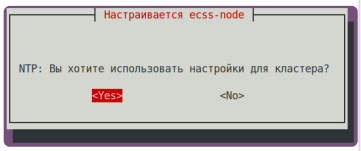
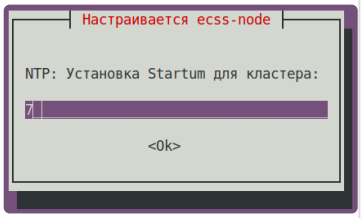
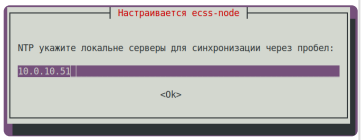
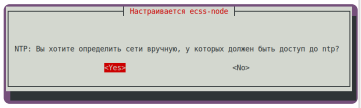
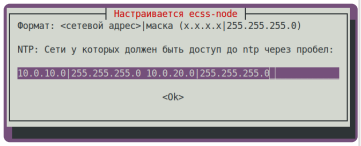
Во время установки будет предложено настроить параметры, необходимые для формирования конфигурационных файлов, примеры ответов следует ниже.

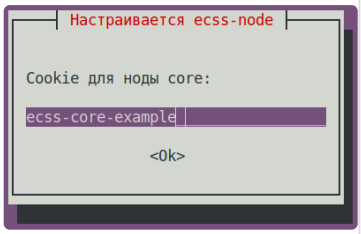
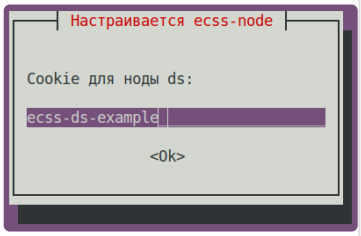
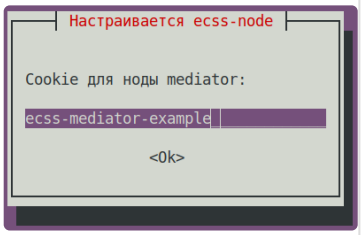
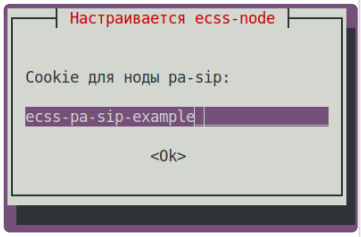
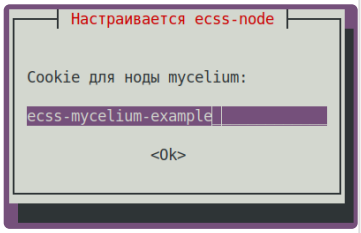
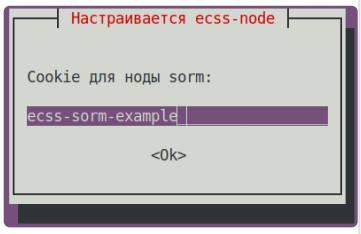
Вопросы ecss-user	Ответы для ecss2	Пример
Хотите ли вы использовать стандартные настройки?	No (значение по умолчанию)	
Хотите ли вы выключить ежедневное обновление apt-пакетов?	Yes (значение по умолчанию)	
Как вы хотите сконфигурировать сертификаты?	**copy** (необходимо выбрать)	
Вы хотите установить сертификаты в систему?	Yes (значение по умолчанию)	
Каким методом вы хотите скопировать сертификаты?	**ssh** (необходимо выбрать)	
Введите имя хоста для подключения:	**ecss1** (необходимо ввести)	
Введите порт хоста для подключения:	22 (значение по умолчанию)	
Введите логин для подключения: (для примера eltex)	**eltex** (необходимо ввести)	

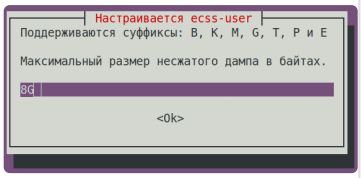
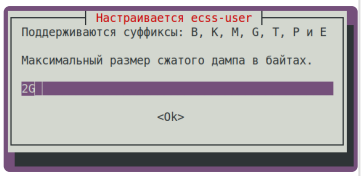
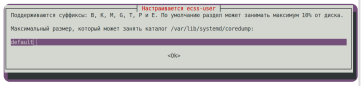
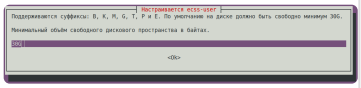
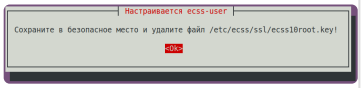


Какой метод авторизации использовать?	**password** (необходимо выбрать)	
Введите пароль для подключения:		
Введите путь до сертификатов:	/etc/ecss/ssl (значение по умолчанию)	

Вопросы ecss-node	**Ответы для ecss2**	**Пример**
Идентификатор ECSS-узла в формате DNS (Настройка имени кластера): (любое уникальное имя, в примере abf.test)	**abf.test** (необходимо ввести) Формат **ECSS_ID** Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире. `[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]`	

Хотите ли вы использовать стандартные настройки?	**No** (необходимо выбрать)	
Выберите пункты, которые хотите настроить:	**ntp + cookie** (необходимо выбрать)	
Внешние NTP серверы через пробел:	ntp.ubuntu.com (значение по умолчанию)	
NTP: Вы хотите использовать настройки для кластера?	Yes (значение по умолчанию)	
NTP: Установка Startum для кластера:	7 (значение по умолчанию)	
NTP укажите локальне серверы для синхронизации через пробел: (в примере ecss2 - 10.0.10.51)	**10.0.10.51** (необходимо ввести)	
NTP: Вы хотите определить сети вручную, у которых должен быть доступ до ntp?	Yes (значение по умолчанию)	
NTP: Сети у которых должен быть доступ до ntp через пробел: (в примере 10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0)	**10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0** (необходимо ввести)	

Cookie для ноды core: (указать уникальный cookie для core , в примере ecss-core-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-core-example** (необходимо ввести)	
Cookie для ноды ds: (указать уникальный cookie для ds , в примере ecss-ds-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-ds-example** (необходимо ввести)	
Cookie для ноды mediator: (указать уникальный cookie для mediator , в примере ecss-mediator-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mediator-example** (необходимо ввести)	
Cookie для ноды pa-sip: (указать уникальный cookie для pa-sip , в примере ecss-pa-sip-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-pa-sip-example** (необходимо ввести)	
Cookie для ноды mycelium: (указать уникальный cookie для mycelium , в примере ecss-mycelium-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mycelium-example** (необходимо ввести)	
Cookie для ноды sorm: (указать уникальный cookie для sorm , в примере ecss-sorm-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-sorm-example** (необходимо ввести)	
Вопросы ecss-user	**Ответы для ecss2**	**Пример**

Максимальный размер несжатого дампа в байтах.	8G (значение по умолчанию)	
Максимальный размер сжатого дампа в байтах.	2G (значение по умолчанию)	
Максимальный размер, который может занять каталог /var/lib/systemd/coredump:	default (значение по умолчанию)	
Минимальный объем свободного дискового пространства в байтах.	30G (значение по умолчанию)	
Сохраните в безопасное место и удалите файл /etc/ecss/ssl/ecss10root.key!	Ok (значение по умолчанию)	

Проверить статус сервисов следующей командой:

на обоих хостах

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
```

В случае статуса "active" продолжаем дальше, в случае статуса "inactive" выполнить команду "sudo systemctl restart ecss-<имя пакета>"

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
active
active
active
active
active
```

8.9.6 Конфигурация NTP

При установке по умолчанию, (Хотите ли вы использовать стандартные настройки?) конфигурация NTP не будет выполнена. При необходимости выполнить конфигурацию NTP:

Конфигурация NTP

Синхронизация времени на серверах

⚠ Перед настройкой NTP нужно убедиться, что в системе установлен пакет **ntp**.

Пример:

```
dpkg -l | grep ntp
ii  ntp                                  1:4.2.8p10+dfsg-5ubuntu7.3
amd64        Network Time Protocol daemon and utility programs
ii  sntp                                 1:4.2.8p10+dfsg-5ubuntu7.3
amd64        Network Time Protocol - sntp client
```

Затем рекомендуется задать значение текущей системной даты, максимально приближенное к реальному времени. Для этого можно воспользоваться утилитой для ручной синхронизации времени **ntpdate**.

Пример для установки времени с сервера ntp.ubuntu.com:

```
sudo ntpdate ntp.ubuntu.com
```

Команда **date** без параметров выводит на экран текущее системное время.

Установка и настройка NTP

Конфигурация NTP настраивается при установке пакета **ecss-node**.

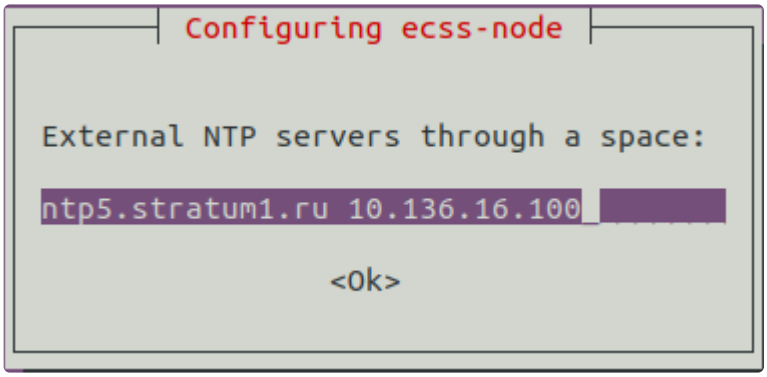
Рассмотрим настройку NTP для кластера из 2-х серверов ecss со следующими параметрами:

Параметр	Значение
Адреса внешних NTP-серверов	• ntp5.stratum1.ru • 10.136.16.100
Локальная синхронизация серверов кластера между собой (orphan mode)	Да, для следующих адресов: • ecss1 - 192.168.1.21 • ecss2 - 192.168.1.22
Подсети, с которых разрешено другим устройствам синхронизироваться с данным сервером	• 192.168.1.0/24 • 10.16.0.0/16

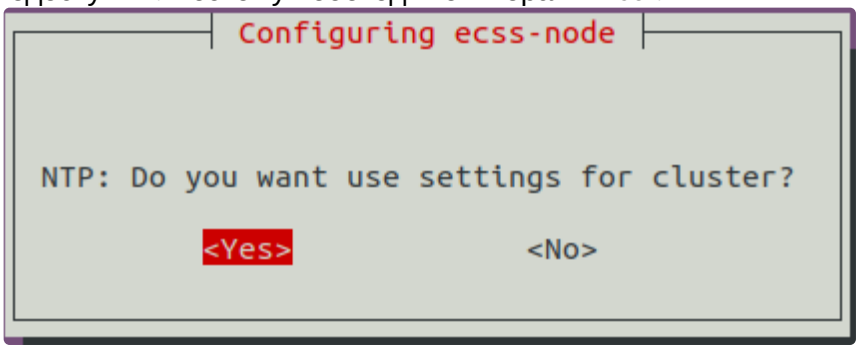
В ходе установки будут заданы несколько вопросов для формирования файла конфигурации.

Ниже приведен пример ответа на вопросы:

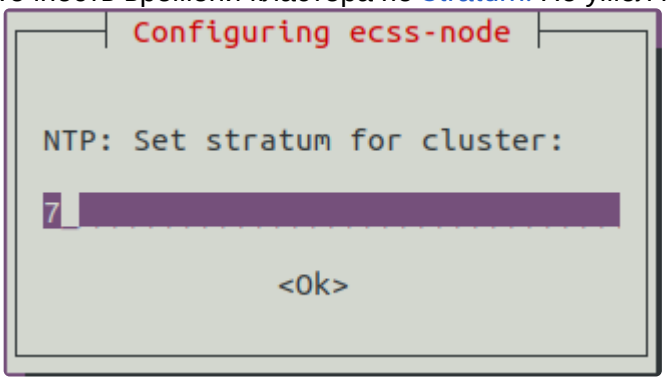
Необходимо ввести внешние сервера через пробел (по умолчанию ntp.ubuntu.com):



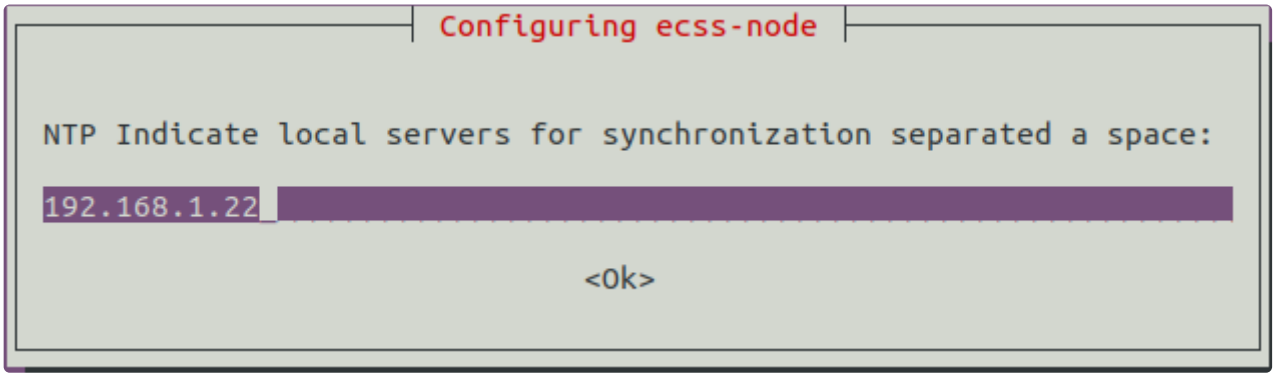
Необходимо разрешить (Yes) или запретить (No) активацию режима tos orphan (режим для кластера, в котором серверы самостоятельно регулируют синхронизацию). Если система устанавливается в кластере, то серверы ECSS должны иметь **одинаковое** время, даже если внешние NTP-серверы недоступны. Поэтому необходимо выбрать "Yes".



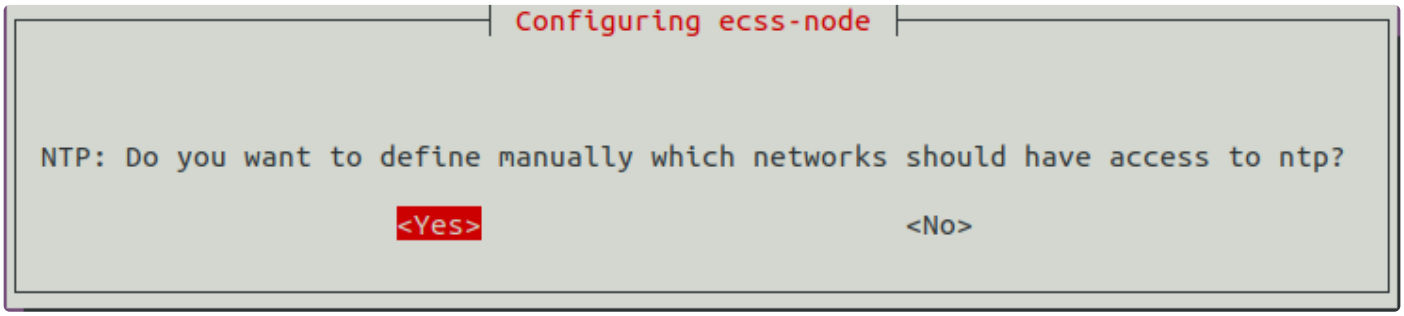
Точность времени кластера по Stratum. По умолчанию — 7:



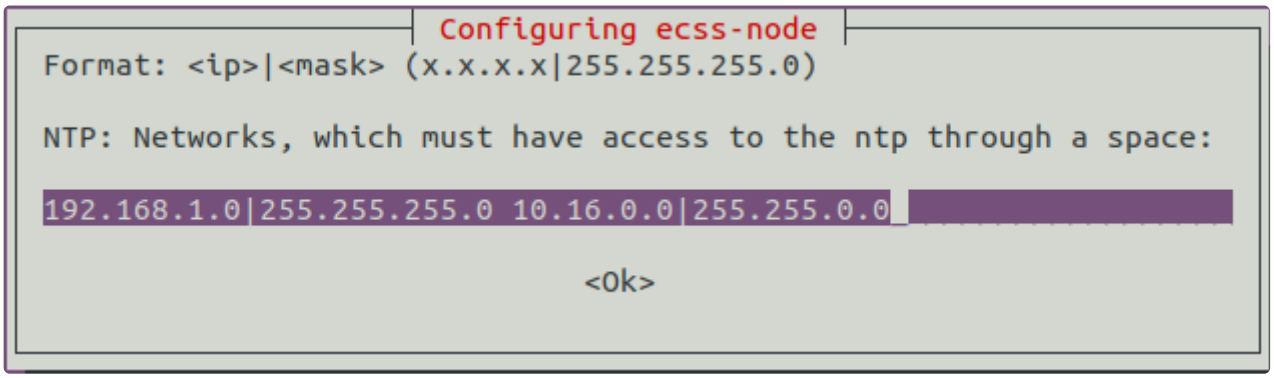
Предлагается ввести адреса соседних серверов кластера для их синхронизации между собой. В данном примере настраиваем ecss1, поэтому вводим адрес ecss2. При настройке ecss2 соответственно вводится адрес ecss1. Если серверов несколько, необходимо перечислить их через пробел.



Далее предлагается настроить адреса подсетей, из которых другим устройствам разрешено синхронизироваться с данным сервером:



Указываются сети, которые могут иметь доступ до данного сервера, чтобы другие ноды, а также прочие устройства могли синхронизировать время с данным сервером. Формат указания сетей: <адрес_сети|маска_сети>. Если сетей несколько, необходимо перечислить их через пробел.



После инсталляции настройки сохраняются в файле **/etc/ecss/ecss-ntp.conf**. Пример получившегося файла для сервера **ecss1**:

```
# /etc/ntp.conf

# http://www.k-max.name/linux/ntp-server-na-linux/
# В preinst делаем резервную копию старого и устанавливаем текущий
# В postrm загружаем из резервной копии

# Смещение системных часов
driftfile       /var/lib/ntp/ntp.drift
# Логи
logfile /var/log/ntp
# Статистика синхронизации времени
statsdir /var/log/ntpstats/

# Разрешает записывать статистику:
# loopstats - статистика для петли (loopback)
# peerstats - статистика пиров
# clockstats - статистика драйвера времени
statistics loopstats peerstats clockstats
filegen loopstats file loopstats type day enable
filegen peerstats file peerstats type day enable
filegen clockstats file clockstats type day enable

# Активация Orphan mode – режима синхронизации времени для кластеров. Устанавливаем
для него stratum (уровень точности: число от 1 до 16)
# tos orphan <stratum>
```

```
# TOS
tos orphan 7 ### INSTALLED AUTOMAT BY ECSS10


# Сервера локальной сети
# peer <ip|domain>
# LOCAL_SERVERS
peer 192.168.1.22 ### INSTALLED AUTOMAT BY ECSS10



# Интернет-сервера
# server xx.xx.xx.xx iburst
# restrict xx.xx.xx.xx
# INTERNET_SERVERS
server ntp5.stratum1.ru iburst ### INSTALLED AUTOMAT BY ECSS10
restrict ntp5.stratum1.ru ### INSTALLED AUTOMAT BY ECSS10
server 10.136.16.100 iburst ### INSTALLED AUTOMAT BY ECSS10
restrict 10.136.16.100 ### INSTALLED AUTOMAT BY ECSS10

# Ограничение доступа к конфигурируемому серверу:
# По умолчанию игнорируем все
restrict -4 default kod notrap nomodify nopeer noquery limited
restrict -6 default kod notrap nomodify nopeer noquery limited
restrict source notrap nomodify noquery

# Localhost без параметров — значит разрешено все. Параметры идут только на
запреты.
restrict 127.0.0.1
restrict ::1
```

Для **ecss2** файл будет аналогичным, за исключением строки пира к соседнему серверу (192.168.1.21):

```
peer 192.168.1.21 ### INSTALLED AUTOMAT BY ECSS10
```

В режиме *Orphan* серверы в кластере синхронизируются друг от друга, сами определяют мастера и следят, чтобы в рамках кластера часы шли синхронно.
Если появляется мастер-сервер NTP со значением stratum меньше заданного для кластера, то кластер автоматически перенастраивается на синхронизацию от него. Таким образом выполняется условие постоянного наличия единственной точки синхронизации времени.

Все зависимые устройства в системе ECSS-10 должны синхронизироваться от серверов кластера. Если используется схема без резервирования, от настройки режима для кластера можно отказаться: тогда в конфигурационном файле будет отсутствовать секция настройки локальных серверов для синхронизации между собой.

⚠ Править конфигурационный файл вручную не рекомендуется, так как при обновлении пакета ecss-node в файл запишутся прошлые настройки из базы debconf от последней реконфигурации пакета.

Правильный способ — использовать команду **dpkg-reconfigure**:

```
sudo dpkg-reconfigure ecss-node
```

Если в файл конфигурации все же были вручную внесены какие-либо изменения, то для применения команды **dpkg-reconfigure** нужно перезапустить сервис NTP:

```
sudo systemctl restart ntp.service
```

Для просмотра информации о состоянии синхронизации используется команда **ntpq –p** . Если использовать дополнительный ключ **–n**, вместо имени сервера будет указан IP-адрес:

Пример:

```
sasha@ecss1:~$ ntpq -p
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
 ecss2           88.147.254.229   2 s   11   64  377    0.099   -1.169   0.357
+10.136.16.100   194.58.204.148   2 u   56  128  377    4.008   -2.482   0.339
*88.147.254.229  .PPS.            1 u  124  128  377   60.440    0.691   0.098
```

Описание параметров:

- remote — имя удаленного NTP-сервера;
- refid — IP-адрес сервера, с которым производит синхронизацию удаленный сервер NTP;
- st — stratum (уровень): число от 1 до 16, отражающее точность сервера;
- t — тип удаленного сервера:
  - u — unicast,
  - l — local,
  - m — multicast,
  - s — symmetric (peer),
  - b — broadcst;
- when — интервал времени (в секундах), прошедший с момента получения последнего пакета от данного сервера;
- poll — интервал между опросами (в секундах), значение варьируется;
- reach — состояние доступности сервера. Восьмеричное представление массива из 8 бит, отражающего результаты последних восьми попыток соединения с сервером. Если последние 8 попыток синхронизации с удаленным сервером были успешны, то параметр принимает значение 377;
- delay — вычисленная задержка ответов от сервера (RTT) в миллисекундах;
- offset — разница между временем локального и удаленного серверов;
- jitter — джиттер, мера статистических отклонений от значения смещения (поле offset) по нескольким успешным параметрам запрос-ответ.

Значение символов перед именами серверов

x — фальшивый источник по алгоритму пересечения;
. — исключён из списка кандидатов из-за большого расстояния;
- — удален из списка кандидатов алгоритмом кластеризации;

+ — входит в конечный список кандидатов;
— выбран для синхронизации, но есть 6 лучших кандидатов;
* — выбран для синхронизации;
o — выбран для синхронизации, но используется PPS;
пробел — слишком большой уровень, цикл или явная ошибка;

После старта сервиса может потребоваться около 10 минут для установления синхронизации времени с базовым NTP-сервером.

Проверить состояние настроенного сервера NTP можно с помощью команды **ntpdate**:

```
sasha@ecss1:~$ sudo ntpdate -q localhost
server 127.0.0.1, stratum 2, offset -0.000032, delay 0.02573
28 Sep 15:00:57 ntpdate[19002]: adjust time server 127.0.0.1 offset -0.000032 sec
```

Как видно, значение stratum сервера стало равно 2.

8.9.7 Установка лицензии

Система ECSS-10 использует ELM-лицензирование – сервис распределения лицензий ECSS-экосистемы, состоит из **ecss-license-agent** и **ecss-license-master** .

- **ECSS License Agent** - Адаптер для работы с EcssLicenseMaster (LM). Получает список хостов EcssLicenseMaster, реализует логику подключения и переподключения к LM.
  - Отвечает за получение, валидацию и дальнейшую отправку лицензий клиенту.
  - Отправляет статусы подключения и полученные лицензии клиенту.
  - Предоставляет API для отправки HTTP запросов и WS событий на LM.
- **Ecss License Master** - Сервис оркестрации параметров лицензии для ECSS-узлов.
  - Отвечает за:
    - Получение лицензии для конкретного ECSS-узла (SSW / AuP) либо устройства (SMG)
    - Распределение параметров лицензии между ECSS-узлами и ECSS-сервисами
    - Контроль активности ECSS-узлов и их валидация во избежание тиражирования лицензии сверх указанных лимитов
    - Контроль изменения указанных лимитов на ECSS-узлах, и их динамическое распределение в соответствии с текущей нагрузкой.

ecss-license-agent является составной частью SSW, входит в пакет ecss-node. Отдельной конфигурации не требует.

В свою очередь **Ecss License Master** может быть установлен как на SSW так и на отдельно выделенный сервер. В случае если Ecss License Master уже установлен на вашем проекте, то потребуется только дополнить запись о новом компоненте в конфигурационном файле LM, и следующий пункт – установка "Установка LM" не потребуется.

Установка LM на SSW host

Для установки пакета **ecss-license-master** необходимо добавить репозиторий следующей командой:

если кластер, то на обоих хостах

```
sudo sh -c "echo deb [arch=amd64] https://archive.eltex.org/ssw/jammy/aup/2.0 stable main extras external >> /etc/apt/sources.list.d/lm-stable.list"
```

Выполнить обновление из добавленного репозитория командой:

если кластер, то на обоих хостах

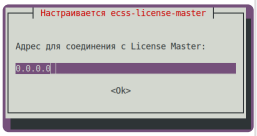
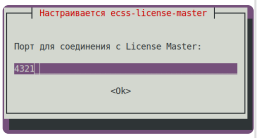
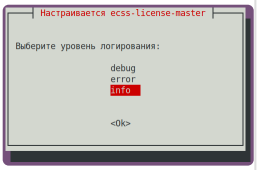
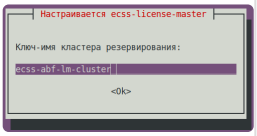
```
sudo apt update
```

Установить пакет LM (ecss-license-master)

если кластер, то на обоих хостах

```
sudo apt install ecss-license-master
```

При установке пакета ecss-license-master будут заданы вопросы:

Вопросы	Ответы	Примеры
Адрес для соединения с License Master: License Master listen ip:	0.0.0.0 (значение по умолчанию)	
Порт для соединения с License Master: License Master listen port:	4321 (значение по умолчанию)	
Выберите уровень логирования: Log level:	info (значение по умолчанию), возможные варианты: debug, error, info	
Ключ-имя кластера резервирования: License Master cluster key:	указать уникальное имя ecss-lm-cluster (значение по умолчанию) Менеджеры с одинаковыми именами попытаются собраться в кластер и резервировать друг друга.	

Данные сохраняются в файле - **/etc/ecss/ecss-license-master/config.env** :

```
ECSS_LM_HOST=0.0.0.0
ECSS_LM_PORT=4321
ECSS_LM_LOG_LEVEL=info
RELEASE_COOKIE=ecss-abf-lm-cluster
```

Конфигурация LM

Сервис ecss-license-master использует два конфигурационных файла: **/etc/ecss/ecss-license-master/config.env** и **/etc/ecss/ecss-license-master/config.yaml**.
В файл **config.env** информация сохраняется в момент установки пакета **ecss-license-master**.
Config.yaml файл содержит основные настройки для подключения к ELM серверу, откуда брать лицензии, какие лицензии, и куда их потом передавать, **все это нужно сконфигурировать вручную**.

При установке LM на SSW

Выполнить команду:

если кластер, то на обоих хостах

```
sudo nano /etc/ecss/ecss-license-master/config.yaml
```

информация в файле после установки

```
elm_addresses: []
licenses: []
ecss_nodes: []
```

В поле **elm_adresses** нужно прописать список серверов ELM к которым мы хотим подключаться для получения лицензий. Первый адрес в списке используется как основной. Дополнительные строки с адресами работают как резерв.

Данные ELM сервера(ов) получить у менеджера проекта.

пример:
- "elm-1.eltex.loc:8099"
- "192.168.111.22:8099"
- "elm-3.eltex.loc:8099"

Дальше задаются сами лицензии в поле **licenses**.

- **id** – может быть любой, главное чтоб был уникальный у каждой лицензии. Влияет только на адрес лицензии на самом LM (например 0,1,2,3).
- **kind** – для SSW поддерживается только ***ssw.***
- **type** – выбирается как ***elm*** если запрашивать данную лицензию нужно с ELM сервера. Если нужно использовать файловую лицензию SSW, то тип выбирается как ***ecss_license***.
    - при ***elm*** типе, необходимо задать поля **license_key** и **product_id** для идентификации лицензии.

- при ***ecss_license*** типе, задается поле **license** содержащее ключ статичной лицензии

⚠ Параметры **license_key** и **product_id** должны быть согласованы с данными загруженными на ELM сервере .

✓ Значение параметров **license_key**, **product_id** или **license**, **passport** нужно получить у менеджера проекта.

Блок **ecss_nodes**.
В нем нужно задать данные сервисов которые будут получать лицензию. В нашем случае SSW.
Сервис задается тремя параметрами:

- **id –** идентификатор сервиса, должен совпадать с ECSS_ID лицензируемого SSW.
- **kind –** тип лицензируемого сервиса. по аналогии с блоком лицензий, для SSW поддерживается только ***ssw*.**
- **license_id –** идентификатор лицензии которая будет отправляться на данный сервис. Соответствует идентификаторам лицензии из блока **licenses**

⚠ Два SSW с **разными ECSS_ID** не могут получить одну и ту же лицензию, конфигурация с таким распределением будет помечена некорректной.
Кроме того, два SSW с **одинаковыми ECSS_ID** не смогут одновременно получать одну и ту же лицензию с одного LM,
поэтому крайне важно корректно устанавливать свой ECSS_ID и делать его уникальным как минимум в зоне видимости используемого LM.

✓ Формат **ECSS_ID**
Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире.

```
[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]
```

⚠ Для SSW работающего в кластере указываем только одну запись в блок **ecss_nodes** на кластер, индивидуально для каждого хоста указывать нельзя. Иначе будут генерироваться два запроса с одинаковым ECSS_ID что недопустимо.

Пример: файла config.yaml

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
   kind: SSW
   type: elm
   license_key: "ssw1test"
   product_id: "ECSS1000001"
 - id: 1
```

```
    kind: ssw
    type: ecss_license
    license: "af615ebb92d381125ff"
ecss_nodes:
 - id: abf.test
    kind: ssw
    license_id: 0
```

Выполнить перезагрузку сервиса **ecss-license-master.service** для обновления конфигурации из файла **/etc/ecss/ecss-license-master/config.yaml** следующей командой:

если кластер, то на обоих хостах

```
sudo systemctl restart ecss-license-master.service
```

При подключении SSW к ранее установленному LM

В случае если LM уже установлен и используется в локальной сети требуется добавить параметры нового SSW в конфигурационный файл **/etc/ecss/ecss-license-master/config.yaml** в блоках licenses и ecss_nodes.

Выполнить команду:

если кластер, то на обоих хостах

```
sudo nano /etc/ecss/ecss-license-master/config.yaml
```

для примера:

Было:

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
    kind: SSW
    type: elm
    license_key: "ssw1test"
    product_id: "ECSS1000001"
 - id: 1
    kind: ssw
    type: ecss_license
    license: "af615ebb92d381125ff"
 - id: 2
    kind: SSW
    type: elm
    license_key: "ssw2test"
    product_id: "ECSS1000002"
ecss_nodes:
 - id: 1.test
```

```
    kind: ssw
    license_id: 0
  - id: 2.test
    kind: ssw
    license_id: 1
  - id: 3.test
    kind: ssw
    license_id: 2
```

Стало:

```
elm_addresses:
  - "elm-1.eltex.loc:8099"
  - "192.168.111.22:8099"
  - "elm-3.eltex.loc:8099"
licenses:
  - id: 0
    kind: SSW
    type: elm
    license_key: "ssw1test"
    product_id: "ECSS1000001"
  - id: 1
    kind: ssw
    type: ecss_license
    license: "af615ebb92d381125ff"
  - id: 2
    kind: SSW
    type: elm
    license_key: "ssw2test"
    product_id: "ECSS1000002"
  - id: 3
    kind: SSW
    type: elm
    license_key: "ssw3test"
    product_id: "ECSS1000003"
ecss_nodes:
  - id: 1.test
    kind: ssw
    license_id: 0
  - id: 2.test
    kind: ssw
    license_id: 1
  - id: 3.test
    kind: ssw
    license_id: 2
  - id: abf.test
    kind: ssw
    license_id: 3
```

Выполнить перезагрузку сервиса **ecss-license-master.service** для обновления конфигурации из файла **/etc/ecss/ecss-license-master/config.yaml** следующей командой:

если кластер, то на обоих хостах

```
sudo systemctl restart ecss-license-master.service
```

Подключение SSW к Менеджера Лицензий

Выполнить конфигурацию подключения SSW к LM следующей командой в CoCon, (в примере LM установлен в кластере, IPadd хостов 10.0.10.51 и 10.0.10.52. значение порта из файла /etc/ecss/ecss-license-master/config.env по умолчанию 4321):

выполняется в CoCon

```
/system/licence/manager/set --hosts [https://10.0.10.51:4321, https://10.0.10.52:4321]
```

В случае установки LM на хостах SSW указывать IP адреса OAM

После выполнения проверить статус подключения командой:

выполняется в CoCon

```
/system/licence/manager/show-status
┌──────────────────────────┬───────┬─────┐
│          Host            │Current│Alive│
├──────────────────────────┼───────┼─────┤
│https://10.0.10.51:4321│*      │true │
│https://10.0.10.52:4321│       │true │
└──────────────────────────┴───────┴─────┘
```

Все хосты должны иметь статус alive=true. Один должен быть "current".

- Current - отображает к какому хосту подключён SSW. Если есть подключение, указывается *, если нет - ничего не указано.
- Alive - отображает статус доступности хоста (healthcheck).

Подключение к License Manager

Отсутствие подключения к License Manager \ ELM будет равноценно отсутствию доступа к токену, что приведёт к критическим авариям, переходу в аварийный режим по истечению которого, если связь не будет восстановлена, система перейдёт на дефолтную лицензию.

данное поведение применимо только при работе в **elm** режиме

Если статус к одному из хостов LM показан current, можно посылать запрос на загрузку лицензии на SSW командой:

выполняется в CoCon

```
/cluster/storage/ds1/licence/request
```

пример выполнения команды /cluster/storage/ds1/licence/request

type=elm

```
/cluster/storage/ds1/licence/request
Licence received
[*******                                                        ] 6s 2ms
Success: Licence parameters applied
```

В режиме "type=ecss_license" команда **/cluster/storage/ds1/licence/request** выполнится только в случае если был предварительно загружен паспорт.

type=ecss_license

```
/cluster/storage/ds1/licence/request
Waiting for licence...
[*******                                                        ] 6s
┌─┬──────────────────────────────────────────────────────┬─────────┬──────────────────────────────┐
│A│                     Description                      │Old Value│              New Value       │
├─┼──────────────────────────────────────────────────────┼─────────┼──────────────────────────────┤
│^│Support for reservation of call-processes             │false    │true                          │
│^│Maximum call duration (in seconds)                    │60       │2678400                       │
│^│Maximum number of simultaneous calls                  │5        │10000                         │
│^│Elph                                                  │         │                              │
│^│ total count of members of the one Elph group         │10       │100                           │
│^│Subscribers limit                                     │10       │500000                        │
│^│Virtual subscribers limit                             │infinity │1000                          │
│^│Add-on conferences                                    │         │                              │
│^│ total count                                          │30       │1000                          │
│^│Add-on conferences                                    │         │                              │
│^│ members of the one add-on conference                 │16       │300                           │
│^│Chat rooms                                            │         │                              │
│^│ total count                                          │30       │100                           │
│^│Chat rooms                                            │         │                              │
│^│ members of the one chatroom                          │16       │300                           │
│^│Meet Me                                               │         │                              │
```

```
|^| total count                                          |0        |100
                |
|^|Meet Me                                              |         |
                |
|^| members of the one Meet Me conference               |0        |300
                |
|^|SORM                                                 |         |
                |
|^| enabled                                             |false    |true
                |
|+| channels on SORM mediator                           |         |64
                |
|^|Sorm extractor                                       |none     |mfi
                |
|^|Support antifraud system                             |none     |custom
                |
|+|Call center                                          |         |
                |
|+| operator's capability for look at calls in queue    |         |true
                |
|+| operator's capability for use Intervension SS       |         |true
                |
|^|Call center                                          |         |
                |
|^| active agents                                       |0        |1000
                |
|^|Call center                                          |         |
                |
|^| active supervisors                                  |0        |100
                |
|^|Teleconference                                       |         |
                |
|^| channels                                            |3        |2000
                |
|^|Teleconference                                       |         |
                |
|^| max members of one teleconference                   |0        |200
                |
|^|Teleconference                                       |         |
                |
|^| active count                                        |0        |32
                |
|^|TSMN system                                          |         |
                |
|^| concurrent calls (active)                           |0        |50
                |
|^|TSMN system                                          |         |
                |
|^| concurrent calls (backup)                           |0        |50
                |
|^|TSMN system                                          |         |
                |
|^| backup mode                                         |none     |"backup"
                |
|^|Total count of simultaneous records voice calls      |0        |200
                |
|+|IVR                                                  |         |
                |
```

```
|+| customization enabled                                  |         |true
              |
|+| Automatic Speech Recognition (ASR) subsystem enabled|         |true
              |
|^|Channels on dialer outgoing calls                      |2        |5
              |
|+|Supplementary Services licence package(s)             |         |
              |
|+| name                                                 |         |'ECSS-ADV'
              |
|+| limit                                                |         |10000
              |
|+| description                                          |         |"Additional services of 4
and 5 levels" |
|+| SS list                                              |         |[1,2,3,4,5]
              |
|+| name                                                 |         |'ECSS-BAS'
              |
|+| limit                                                |         |10000
              |
|+| description                                          |         |"Basic services of 1 and
2 levels"     |
|+| SS list                                              |         |[1,2]
              |
|+| name                                                 |         |'ECSS-BAS+'
              |
|+| limit                                                |         |10000
              |
|+| description                                          |         |"Expansion of basic
services of 3 level"|
|+| SS list                                              |         |[1,2,3]
              |
|^|GEO backup                                            |         |
              |
|^| enabled                                              |false    |true
              |
|^|SIGTRAN                                               |         |
              |
|^| SEP                                                  |false    |true
              |
|^| STP                                                  |false    |true
              |
└─┴──────────────────────────────────────────────────────┴─────────┴──────────────────────
──────────────┘
Legend:
    '^' - Changed;
    '-' - Deleted;
    '+' - New.


[request] You are trying to add a license that is different from the current one by the pa
          rameters in the table (other values will remain unchanged).
Licence received      o ?> yes
[*******                                                    ] 1m 15s
Success: Licence parameters applied
```

В зависимости от выбранного типа лицензии для SSW : "type=**elm**" или "type=**ecss_license**" на команду просмотра лицензий получим разный вывод:

type=ecss_license

```
/cluster/storage/ds1/licence/list-licence
┌──┬────────────────────┬──────────┬──────┬──────────────────┬────────────────────┬──────────────────┐
│Id│Creation date(UTC) │   SSW ID   │Active│  Description   │Comm. Exp.
Date(UTC)│Expiration date(UTC)│Time left│
├──┼────────────────────┼──────────┼──────┼──────────────────┼────────────────────┼──────────────────┤
│1 │12.03.2026 18:40:00│ECSS 010070 │*     │ECSS TPM License│13.03.2027 06:59:59
│14.03.2026 10:44:05 │1d 6h 59m│
│0 │01.01.1990 00:00:00│ECSS DEFAULT│      │Default licence │                    │
│          │
└──┴────────────────────┴──────────┴──────┴──────────────────┴────────────────────┴──────────────────┘
```

TPM - является полным аналогом лицензирования которое использовалось раньше, с токеном и паспортом, без подключения к LM.

или

type=elm

```
/cluster/storage/ds1/licence/list-licence
┌──┬────────────────────┬──────────┬──────┬──────────────────┬────────────────────┬──────────────────┐
│Id│Creation date(UTC) │   SSW ID   │Active│  Description   │Comm. Exp.
Date(UTC)│Expiration date(UTC)│Time left│
├──┼────────────────────┼──────────┼──────┼──────────────────┼────────────────────┼──────────────────┤
│1 │03.02.2026 16:26:34│ECSS2000009 │*     │ECSS ELM License│03.02.2027 16:26:34
│14.03.2026 10:49:12 │1d 6h 59m│
│0 │01.01.1990 00:00:00│ECSS DEFAULT│      │Default licence │                    │
│          │
└──┴────────────────────┴──────────┴──────┴──────────────────┴────────────────────┴──────────────────┘
```

8.9.8 Установка остальных пакетов ecss и их настройка

Выполнить конфигурацию для Sip адаптера , выполнив следующую команду:

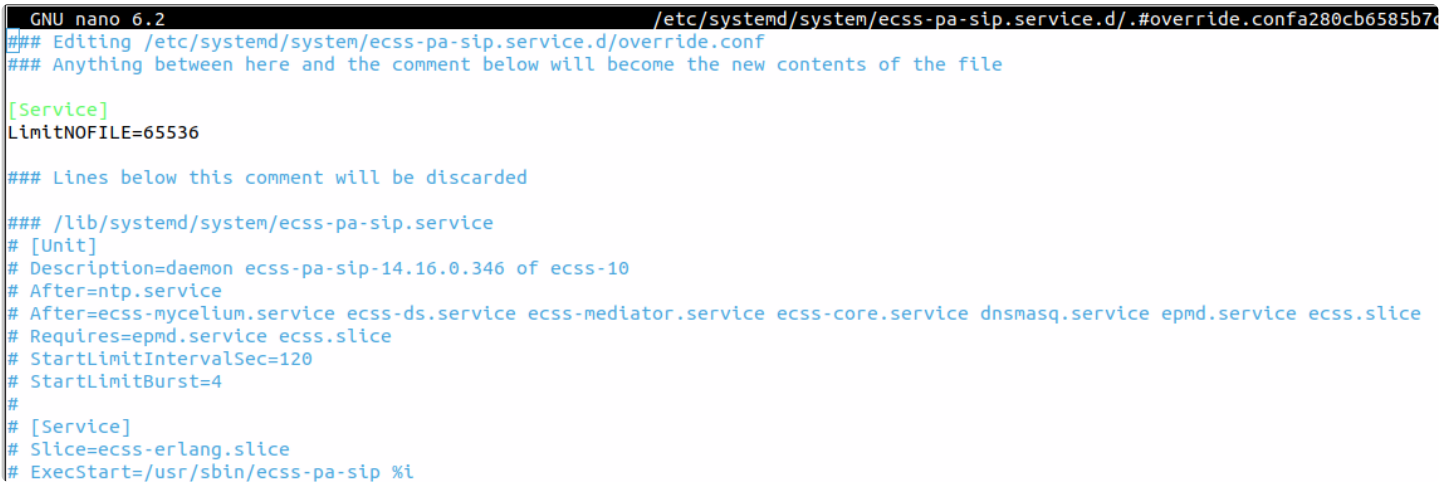
на обоих хостах

```
sudo systemctl edit ecss-pa-sip.service
```

Добавить в конфигурационный файл следующие данные:

```
[Service]
LimitNOFILE=65536
```

Вставить эти две строчки необходимо в определенное место, (начиная с Ubuntu 22.04.2 есть несколько важных моментов. Файл, который открывается, полностью закомментирован, но первые несколько строк выглядят так (см рисунок ниже). Любые вносимые вами правки должны быть между этими двумя блоками комментариев.)



Перезагрузите конфигурацию

на обоих хостах

```
sudo systemctl daemon-reload
```

Актуализировать конфигурационный файл, выполнив перезагрузку сервера:

на обоих хостах

```
sudo systemctl restart ecss-mycelium ecss-ds ecss-core ecss-pa-sip ecss-mediator
```

Для корректной работы кластерной конфигурации необходимо настроить работу **RestFS** на базе **GlusterFS-сервера.**

Следует проверить в файле /etc/dnsmasq.d/ecss-broker, что адреса **primary и secondary broker** соответствуют указанным при инсталляции пакета **ecss-node**.

на обоих хостах

```
cat /etc/dnsmasq.d/ecss-broker
```

Пример содержания файла на **ecss1** и **ecss2** (содержимое файлов должно быть **одинаковым** на обоих серверах):

```
address=/primary.broker.ecss/10.0.10.51
address=/secondary.broker.ecss/10.0.10.52
```

glusterfs

Выполните настройку glusterfs для ecss-restfs на **первом хосте (ecss1)**,
для этого **установите** пакеты glusterfs-server и attr **на обоих** хостах:

на обоих хостах

```
sudo apt install -y glusterfs-server attr
```

Запустите glasterd на **обоих** хостах кластера:

на обоих хостах

```
sudo systemctl start glusterd
```

После запуска выполните создание связи с удаленным виртуальным хостом, выполните команду на **ecss1**:

на ecss1

```
sudo gluster peer probe 10.0.10.52
```

Проверьте наличие созданной связи, выполните команду на **ecss2**:

на ecss2

```
sudo gluster peer status
```

Должна появиться информация о первом хосте — ecss1:

```
Number of Peers: 1

Hostname: ecss1
Uuid: d7319720-ea3a-43a4-a7d5-e97bf9d205b9
State: Peer in Cluster (Connected)
```

Выполните аналогичную проверку на другом хосте — **ecss1**. Вы также должны увидеть информацию о партнере **ecss2** (или его IP-адрес).

Создайте кластер для репликации, запустите репликацию и посмотрите ее статус, выполните следующие команды на **ecss1**:

на ecss1

```
sudo gluster volume create ecss_volume replica 2 transport tcp 10.0.10.51:/var/lib/ecss/
glusterfs 10.0.10.52:/var/lib/ecss/glusterfs force
```

На **ecss1**:

на ecss1

```
sudo gluster volume start ecss_volume
```

На **обоих** хостах **ecss1** и **ecss2**:

на обоих хостах

```
sudo gluster volume info
```

Таким образом, статус репликации будет выглядеть следующим образом, необходимо обратить внимание на поля "**Status**" и "**Bricks**" — они должны иметь следующий вид:

```
Volume Name: ecss_volume
Type: Replicate
Volume ID: 3bfc7587-0f85-48ed-9612-21f0d79c6e52
Status: Started
Snapshot Count: 0
Number of Bricks: 1 x 2 = 2
Transport-type: tcp
Bricks:
Brick1: 10.0.10.51:/var/lib/ecss/glusterfs
Brick2: 10.0.10.52:/var/lib/ecss/glusterfs
Options Reconfigured:
transport.address-family: inet
nfs.disable: on
performance.client-io-threads: off
```

Чтобы смонтировать glusterfs раздел, выполните на **обоих** хостах **ecss1** и **ecss2** следующие действия:

- Создайте новый systemd unit

на обоих хостах

```
sudo nano /etc/systemd/system/ecss-glusterfs-mount.service
```

- Добавьте в него следующую конфигурацию:

```
[Unit]
Description=mount glusterfs
After=network.target
Requires=network.target

[Service]
RemainAfterExit=no
Type=forking
RestartSec=10s
Restart=always
ExecStart=/sbin/mount.glusterfs localhost:/ecss_volume /var/lib/ecss/restfs -o fetch-
attempts=10
ExecStop=/bin/umount /var/lib/ecss/restfs

[Install]
WantedBy=multi-user.target
```

- Добавить unit в автозагрузку на **обоих** хостах.

на обоих хостах

```
sudo systemctl enable glusterd.service
```

на обоих хостах

```
sudo systemctl enable ecss-glusterfs-mount.service
```

- Создайте директорию для корректной установки RestFS на **обоих** хостах:

на обоих хостах

```
sudo mkdir /var/lib/ecss/restfs
```

Выполнить следующие команды на **обоих** хостах:

на обоих хостах

```
sudo systemctl daemon-reload
```

на обоих хостах

```
sudo systemctl restart ecss-glusterfs-mount.service
```

Проверьте, что раздел подмонтировался, для этого на обоих хостах выполните команду - **df -h**:

на обоих хостах

```
df -h
```

```
/dev/sda8                      4,5G  213M  4,1G   5% /var/log
/dev/sda5                       37G   48M   35G   1% /var/lib/ecss/ecss-media-server/records
/dev/sda6                       19G   44M   18G   1% /var/lib/ecss/cdr
/dev/sda7                       19G   44M   18G   1% /var/lib/ecss/statistics
/dev/sda9                       19G  7,6G  9,7G  44% /var/log/ecss
localhost:/ecss_volume          46G   59M   44G   1% /var/lib/ecss/restfs
```

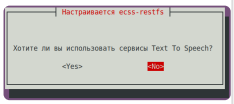
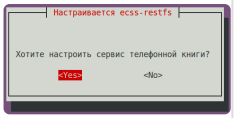
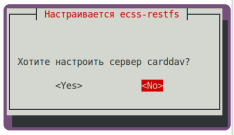
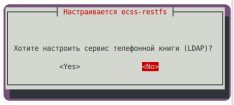
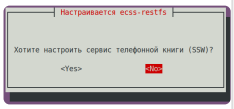
При просмотре информации должен появиться подмонтированный раздел — **localhost:/ecss_volume** 46G 59M 44G 1% **/var/lib/ecss/restfs**

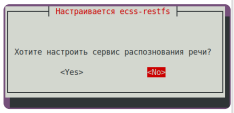
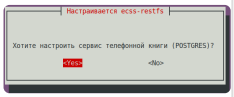
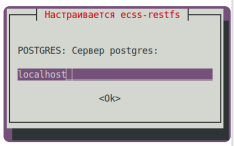
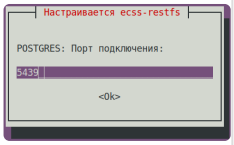
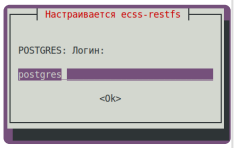
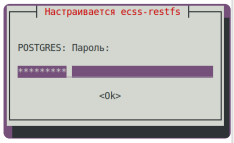
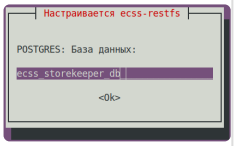
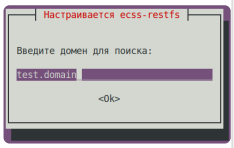
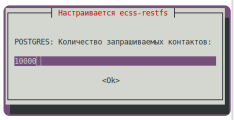
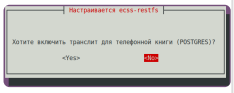
Установка пакета ecss-restfs и его настройка

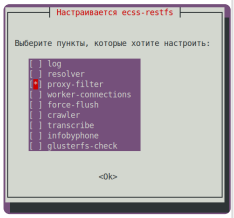
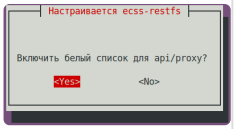
на обоих хостах

```
sudo apt install -y ecss-restfs
```

Установка **ecss-restfs.** В ходе установки нужно будет ответить на ряд вопросов для создания необходимых конфигурационных файлов. Также инсталлятор предложит установить и настроить пакет Text2speech от Yandex.

Вопросы ecss-restfs	Ответы	Пример
Хотите ли вы использование функции Text To Speeh (Use TTS service)	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (Configure phone book)	**Yes** (нужно указать)	
Хотите настроить сервер carddav?	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (LDAP)?	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (SSW)?	No (значение по умолчанию)	

Вопросы ecss-restfs	Ответы	Пример
Хотите настроить сервис распознавания речи (Configure speech recognition service)	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (POSTGRES)?	Yes (значение по умолчанию)	
POSTGRES: Сервер postgres:	localhost (значение по умолчанию)	
POSTGRES: Порт подключения:	5439 (значение по умолчанию)	
POSTGRES: Логин:	postgres (значение по умолчанию)	
POSTGRES: Пароль:	postgres1 (значение по умолчанию)	
POSTGRES: База данных:	ecss_storekeeper_db (значение по умолчанию)	
Введите домен для поиска:	test.domain (значение по умолчанию)	
POSTGRES: Количество запрашиваемых контактов:	10000 (значение по умолчанию)	
Хотите включить транслит для телефонной книги (LDAP)?	No (значение по умолчанию)	

Вопросы ecss-restfs	Ответы	Пример
Выберите пункты, которые хотите настроить:	**proxy-filter** (необходимо выбрать)	
Включить белый список для api/proxy?		

белый список адресов прописать в файл /etc/ecss/ecss-restfs/api-proxy.conf

```
# ------------------------------------------------------------
# Формат файла фильтра API (api/proxy):
#   - IP: 192.168.0.1
#   - Домен: example.com
#   - Wildcard: *.wildcard.com
#   - Комментарии: строки, начинающиеся с #
# Пример:
#   192.168.0.1
#   example.com
#   # комментарий
#   *.wildcard.com
# ------------------------------------------------------------
```

ВАЖНО!

В случае успешного проксирование (200) кеш хранится 28 дней, поэтому для применения конфигурации фильтра, в случае изменения, - кеш (/var/cache/ecss/restfs-api) стоит удалить и перезапустить сервис ecss-restfs.

после установки пакета ecss-restfs проверьте наличие wav файлов авто-информатора в директории /var/lib/ecss/restfs/system/sounds/ командой:

на обоих хостах

```
ll /var/lib/ecss/restfs/system/sounds/
```

наличие wav файлов в указной выше директории говорит о корректности установки пакета ecss-restfs.

Отсутствия wav файлов в указной выше директории, говорит о проблеме при установки пакета ecss-restfs. Для ее решения выполните команду:

```
sudo apt --fix-broken install
```

и повторите команду проверки:

```
ll /var/lib/ecss/restfs/system/sounds/
```

Осталось проверить доступность этих файлов извне, для этой проверки выполним поочередно команду на хосте ecss1 и ecss2:

```
wget http://ecss1:9990/system/sounds/ai_you.wav
```

```
wget http://ecss2:9990/system/sounds/ai_you.wav
```

```
wget http://ecss1:9990/system/sounds/ai_you.wav
--2023-12-18 17:43:29--  http://ecss1:9990/system/sounds/ai_you.wav
Resolving ecss1 (ecss1)... 127.0.1.1, 10.0.10.51
Connecting to ecss1 (ecss1)|127.0.1.1|:9990... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11670 (11K) [audio/x-wav]
Saving to: ‘ai_you.wav’

ai_you.wav                                          100%
[=================================================================================
===>]  11,40K  --.-KB/s    in 0s

2023-12-18 17:43:29 (301 MB/s) - ‘ai_you.wav’ saved [11670/11670]
```

проверочные файлы можно удалить

```
rm ai_you.wav
rm ai_you.wav.1
```

если SSW работает в кластере

В случае кластерной конфигурации, для предотвращения одновременной обработки/конвертации/удаления одного файла (.pcm/.wav/.mp3) на разных серверах ecss1/ecss2 → сервис **ecss-restfs-crawler** должен быть запущен **только на одном из серверов** кластера. Для этого на сервере ecss2 выполнить следующие команды:

на ecss2

```
sudo systemctl disable ecss-restfs-crawler.timer
```

на ecss2

```
sudo systemctl stop ecss-restfs-crawler.timer
```

ecss-media-server

на обоих хостах

```
sudo apt install -y ecss-media-server
```

Для медиасервера (*ecss-media-server/MSR*) возможно начальное конфигурирование с записью параметров в файл конфигурации,
для этого нужно провести конфигурацию **transport bind-addr,mcc bind-addres**:

Вопросы ecss-media-server	Ответы для ecss1	Ответы для ecss2
[MSR SIP] Введите bind-ip адрес (Enter)	**10.0.20.51** (необходимо ввести)	**10.0.20.52** (необходимо ввести)
[MSR Control-Channel] Введите bind-ip-адрес	**10.0.20.51** (необходимо ввести)	**10.0.20.52** (необходимо ввести)

После формирования конфигураций по умолчанию производим проверку:

```
cat /etc/ecss/ecss-media-server/config.xml
```

Внутри лежит конфигурация для msr: config.xml, в директории conf.d лежит конфигурация default.xml. По своей сути default.xml — это дополнение config.xml, которое определяет секцию аккаунтов. Это сделано для того, чтобы после обновлений пакета данная конфигурация оставалась неизменной. Вид config.xml представлен здесь: файл конфигурации.

```
<?xml version="1.0" encoding="utf-8"?>
<config date="02:30:33 07.12.2023">
  <general log-level="3" log-rotate="yes" max-calls="8192" max-in-group="512" load-sensor="medi
a" load-delta="10" calls-delta="100" spool-dir-size="100M" log-name="msr.log" log-path="/var/
log/ecss/media-server" use-srtp="disabled" enable-ice-transport="no" ice-update="no"
 aggressive-ice="yes" stun-server="" suspicious-mode="no"/>
  <transport bind-addr="10.0.20.51" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca" user="webrtc@live.com" password="muazkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60" rtcp-timeout="0" rtp-
timeout="350" udp-src-check="no" cn-multiplier="3" port-start="12000" port-range="2048" tias-
in-sdp="no" thread-cnt="2" vid-enc-threads="2" vid-dec-threads="2" video-conf-layout="evenly"
 keyframe-interval="1000" vid-decode-delay="100" silence-threshold="-30" dtmf-flash-disable="no
" video-dscp="0" other-dscp="0" dummy-video-src="/usr/share/ecss-media-server/video/
dummy_video.yuv" video-enc-width="640" video-enc-height="360" finalsilence="1000" rtcp-stat-
dump="yes" silent-codec-switch="yes"/>
  <codec pcma="1" pcmu="2" ilbc="0" gsm="0" g722="3" g726="0" g729="0" speex="0" l16="0" g7221="
0" opus="0" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
  <accounts>
    <!-- <dynamic msr_name="msr.name"
            realm="sip:127.0.0.1:5000"
            dtmf_mode="rfc+inband+info"
            auth_name="user"
            auth_password="password" /> -->
  </accounts>
  <pbyte>
```

```
    <mcc bind-addr="10.0.20.51" port="5700"/>
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

По умолчанию после инсталляции ECSS-10 **активизирует только** кодеки — **pcma, pcmu, g722,** h264, h263-1998, t38. Остальные имеют статус =0 = выкл. Если хотите активизировать остальные кодеки, в редакторе nano измените приоритет с 0 на цифру приоритета (уровень приоритета не должен повторятся для сервиса audio/video/fax)
sudo nano /etc/ecss/ecss-media-server/config.xml

чтобы активизировать **все** кодеки
строчку -
<codec pcma="1" pcmu="2" ilbc="**0**" gsm="**0**" g722="3" g726="**0**" g729="**0**" speex="**0**" l16="**0**" g7221="**0**" opus="**0**" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
привести к виду -
<codec pcma="1" pcmu="2" ilbc="**6**" gsm="**4**" g722="3" g726="**11**" g729="**5**" speex="**7**" l16="**8**" g7221="**9**" opus="**10**" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>

Недопустимо указывать у разных кодеков одинаковый приоритет, для примера g726="5" g729="5". В этом случае они будут работать не корректно/игнорироваться.
За исключением "0" - выключено.

Рассмотрим секцию аккаунтов (файл default.xml):

```
cat /etc/ecss/ecss-media-server/conf.d/default.xml
```

После инсталляции пакетов в фале будет указан ip-add 127.0.0.1 , и только для одного хоста.

```
<?xml version="1.0"?>
<config>
      <accounts>
            <dynamic msr_name="msr.ecss1" realm="sip:127.0.0.1:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>
      </accounts>
</config>
```

Требуется указать в конфигурации для обоих хостов, реальные IP-add.

после инсталляции пакетов в фале будет указан только свой локальный ip-add , необходимо добавить строчку с ip-add партнера и привести содержимое файла к виду указанному снизу:

Необходимо привести содержимое файла к виду указанному снизу (в примере ecss1 имеет IP-add 10.0.20.51/ ecss2 - 10.0.20.52):

на обоих хостах

```
sudo nano /etc/ecss/ecss-media-server/conf.d/default.xml
```

Настройка msr для ecss1 (/etc/ecss/ecss-media-server/conf.d/default.xml)

```
<?xml version="1.0"?>
<config>
    <accounts>
        <dynamic msr_name="msr.ecss1"
 realm="sip:10.0.20.51:5000" dtmf_mode="rfc+inband+info" auth_name="user"
 auth_password="password">
            <via dynamic-ifaces="no">
                <iface name="net.20" ip="10.0.20.51"/>
            </via>
        </dynamic>
        <dynamic msr_name="msr.ecss1"
 realm="sip:10.0.20.52:5000" dtmf_mode="rfc+inband+info" auth_name="user"
 auth_password="password">
            <via dynamic-ifaces="no">
                <iface name="net.20" ip="10.0.20.52"/>
            </via>
        </dynamic>
    </accounts>
</config>
```

Настройка msr для ecss2 (/etc/ecss/ecss-media-server/conf.d/default.xml)

```
<?xml version="1.0"?>
<config>
    <accounts>
        <dynamic msr_name="msr.ecss2" realm="sip:10.0.20.52:5000" dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password">
            <via dynamic-ifaces="no">
                <iface name="net.20" ip="10.0.20.52"/>
            </via>
        </dynamic>
        <dynamic msr_name="msr.ecss2" realm="sip:10.0.20.51:5000" dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password">
            <via dynamic-ifaces="no">
                <iface name="net.20" ip="10.0.20.51"/>
            </via>
        </dynamic>
    </accounts>
</config>
```

В ней указаны текущие настройки, согласно которым происходит регистрация msr на core.

Основными параметрами здесь являются: msr_name и realm:

- msr_name — параметр, определяющий название msr. (рекомендуется задавать msr. и к какому хосту он принадлежит, например msr.ecss1);
- realm — определяет адрес для регистрации на ядре. Точка входа по умолчанию — порт 5000.

После изменения конфигурации файлов **/etc/ecss/ecss-media-server/config.xml** и **/etc/ecss/ecss-media-server/conf.d/default.xml** необходимо выполнить перезапуск сервиса **ecss-media-server**, чтобы изменения вступили в силу, следующей командой:

на обоих хостах

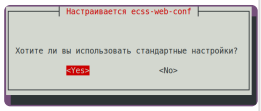
```
sudo systemctl restart ecss-media-server
```

ecss-web-conf

на обоих хостах

```
sudo apt install -y ecss-web-conf
```

Web-конфигуратор позволяет сделать управление системой более наглядным и комфортным. Установка web-конфигуратора не является обязательной, но рекомендуется.
Также при установке пакета **ecss-web-conf** автоматически устанавливается пакет **ecss-subsriber-portal-ui**. Приложение "Портал абонента" системы ECSS-10 позволяет абонентам системы самостоятельно управлять услугами, просматривать информацию по совершенным вызовам, активным конференциям, а также настраивать собственные IVR-скрипты для входящих вызовов. Описание работы веб-конфигуратора приведено в разделе "Портал абонента".

Вопросы ecss-web-conf	Ответы для обоих хостов	Пример
Хотите ли вы использовать стандартные настройки?	Yes (значение по умолчанию)	

Проверьте возможность подключение к web интерфейсу http://10.0.20.51, пользователь/пароль - admin/password, в настоящий момент не все функции будут доступны, но web интерфейс должен быть рабочий и логин успешный.

8.9.9 Настройка VRRP для SIP-адаптера

Выполните настройку VRRP для SIP-адаптера, для этого на **обоих** хостах создайте файлы следующего содержания:

Глобального конфигурационного файла **keepalived.conf**. На обоих хостах содержимое файла **одинаковое**:

на обоих хостах

```
sudo nano /etc/keepalived/keepalived.conf
```

скопируйте и вставьте следующий текст в файл /etc/keepalived/keepalived.conf

на обоих хостах

```
global_defs {
    vrrp_version 3          # версия протокола VRRP (2 или 3)
    script_user nobody      # ограниченный в правах системный пользователь, от которого будут
запускаться скрипты проверки доступности
    enable_script_security  # не запускать скрипты от root, если часть пути к ним доступна на
запись для обычных пользователей
}

include /etc/keepalived/sip.conf
```

Конфигурационного файла sip.conf:

на обоих хостах

```
sudo nano /etc/keepalived/sip.conf
```

etc/keepalived/sip.conf для ecss1

```
vrrp_script check_sip {
    script "/usr/bin/ecss_pa_sip_port
65535"
    interval 2
    timeout 2
}

# Конфигурация адреса для первого
виртуального адреса SIP-адаптера
vrrp_instance SIP1 {
    state MASTER                    #
Исходное состояние при старте
    interface net.20                # Имя
сетевого интерфейса, на котором будет
работать протокол VRRP
    virtual_router_id 31            #
Уникальный идентификатор роутера
(0..255)
    priority 100                    #
Приоритет (0..255) чем выше - тем
больше
    advert_int 1                    #
Интервал рассылки уведомлений (с)
    preempt_delay 60                #
Интервал ожидания мастера при старте
демона (с) при исходном состоянии
BACKUP

    unicast_src_ip 10.0.20.51     #
Собственный реальный IP-адрес
    unicast_peer {
        10.0.20.52                  #
Реальный IP-адрес соседа
    }

    virtual_ipaddress {
        # Виртуальный IP-адрес и маска
        # dev - сетевой интерфейс, на
котором будет поднят виртуальный адрес
        # label - метка виртуального
интерфейса (для удобства идентификации)
        10.0.20.31/24 dev net.20 label
net.20:SIP1
    }
```

etc/keepalived/sip.conf для ecss2

```
vrrp_script check_sip {
    script "/usr/bin/ecss_pa_sip_port
65535"
    interval 2
    timeout 2
}

# Конфигурация адреса для первого
виртуального адреса SIP-адаптера
vrrp_instance SIP1 {
    state BACKUP                    #
Исходное состояние при старте
    interface net.20                # Имя
сетевого интерфейса, на котором будет
работать протокол VRRP
    virtual_router_id 31            #
Уникальный идентификатор роутера
(0..255)
    priority 50                     #
Приоритет (0..255) чем выше - тем
больше
    advert_int 1                    #
Интервал рассылки уведомлений (с)
    preempt_delay 60                #
Интервал ожидания мастера при старте
демона (с) при исходном состоянии
BACKUP

    unicast_src_ip 10.0.20.52     #
Собственный реальный IP-адрес
    unicast_peer {
        10.0.20.51                  #
Реальный IP-адрес соседа
    }

    virtual_ipaddress {
        # Виртуальный IP-адрес и маска
        # dev - сетевой интерфейс, на
котором будет поднят виртуальный адрес
        # label - метка виртуального
интерфейса (для удобства идентификации)
        10.0.20.31/24 dev net.20 label
net.20:SIP1
    }
```

etc/keepalived/sip.conf для ecss1

```
    track_script {
        check_sip
    }
}

# Конфигурация адреса для второго
виртуального адреса SIP-адаптера

vrrp_instance SIP2 {
    state BACKUP                        #
Исходное состояние при старте
    interface net.20                    #
Имя сетевого интерфейса, на котором
будет работать протокол VRRP
    virtual_router_id 32                #
Уникальный идентификатор роутера
(0..255)
    priority 50                         #
Приоритет (0..255) чем выше - тем
больше
    advert_int 1                        #
Интервал рассылки уведомлений (с)
    preempt_delay 60                    #
Интервал ожидания мастера при старте
демона (с) при исходном состоянии
BACKUP

    unicast_src_ip  10.0.20.51        #
Собственный реальный IP-адрес
    unicast_peer {
         10.0.20.52                     #
Реальный IP-адрес соседа
    }

    virtual_ipaddress {
        # Виртуальный IP-адрес и маска
        # dev - сетевой интерфейс, на
котором будет поднят виртуальный адрес
        # label - метка виртуального
интерфейса (для удобства идентификации)
        10.0.20.32/24 dev net.20 label
net.20:SIP2
    }

}
```

etc/keepalived/sip.conf для ecss2

```
}

# Конфигурация адреса для второго
виртуального адреса SIP-адаптера

vrrp_instance SIP2 {
    state MASTER                        #
Исходное состояние при старте
    interface net.20                    #
Имя сетевого интерфейса, на котором
будет работать протокол VRRP
    virtual_router_id 32                #
Уникальный идентификатор роутера
(0..255)
    priority 100                        #
Приоритет (0..255) чем выше - тем
больше
    advert_int 1                        #
Интервал рассылки уведомлений (с)
    preempt_delay 60                    #
Интервал ожидания мастера при старте
демона (с) при исходном состоянии
BACKUP

    unicast_src_ip  10.0.20.52        #
Собственный реальный IP-адрес
    unicast_peer {
         10.0.20.51                     #
Реальный IP-адрес соседа
    }

    virtual_ipaddress {
        # Виртуальный IP-адрес и маска
        # dev - сетевой интерфейс, на
котором будет поднят виртуальный адрес
        # label - метка виртуального
интерфейса (для удобства идентификации)
        10.0.20.32/24 dev net.20 label
net.20:SIP2
    }

    track_script {
        check_sip
    }
}
```

В данном случае была добавлена поддержка виртуальных интерфейсов со связью Master - Backup. Для ***ecss1*** основным будет являться ***net.20:SIP1,*** а запасным — ***net.20:SIP2*** соответственно. Важно учесть, что

в конфигурации учтено использование адресных переменных. В секции interface требуется указать, на каком интерфейсе будут слушаться VRRP-сообщения, а в секции virtual_ipaddress требуется указать интерфейс, на котором будет восстановлен виртуальный адрес.

Выполните перезапуск keepalived:

на обоих хостах

```
sudo systemctl restart keepalived.service
```

для проверки правильности установки выполните команду ifconfig, в конце выходного сообщения вы должны видеть новый интерфейс - net.20:SIP1 с выбранным IP-адресом

```
ifconfig

. . .
net.20:SIP1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 10.0.20.31 netmask 255.255.255.0 broadcast 0.0.0.0
ether 08:00:27:6d:4f:02 txqueuelen 1000 (Ethernet)

ifconfig

. . .
net.20:SIP2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.20.32  netmask 255.255.255.0  broadcast 0.0.0.0
        ether 08:00:27:05:03:39  txqueuelen 1000  (Ethernet)
```

8.9.10 Дальнейшая настройка

snpmd

Настройка SNMP описана в разделе "Управление и мониторинг по протоколу SNMP".

8.9.11 Настройка безопасности. SSH

Выполните настройку сервера SSH:

на обоих хостах

```
sudo nano /etc/ssh/sshd_config
```

Укажите в конфигурационном файле порт и адрес, по которому можно обратиться к серверу:

Настройка ssh для ecss1 (/etc/ssh/sshd_config)	Настройка ssh для ecss2 (/etc/ssh/sshd_config)
# This is the sshd server system-wide configuration file. See # sshd_config(5) for more information. # This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin # The strategy used for options in the default sshd_config shipped with # OpenSSH is to specify options with their default value where # possible, but leave them commented. Uncommented options override the # default value. **Port 2000** #AddressFamily any **ListenAddress 10.0.10.51** #ListenAddress :: <...>	# This is the sshd server system-wide configuration file. See # sshd_config(5) for more information. # This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin # The strategy used for options in the default sshd_config shipped with # OpenSSH is to specify options with their default value where # possible, but leave them commented. Uncommented options override the # default value. **Port 2000** #AddressFamily any **ListenAddress 10.0.10.52** #ListenAddress :: <...>

Перезапустите ssh:

на обоих хостах

```
sudo systemctl restart ssh.service
```

Зайдите в CLI - CoCon:

на ecss1 пароль - password

```
ssh admin@localhost -p 8023
```

8.9.12 Добавление услуг

Установка всех услуг указных в лицензии:

выполняется в CoCon

```
/cluster/storage/ds1/ss/install ds1@ecss1 ss_*
```

```
/cluster/storage/ds1/ss/install ds1@ecss1 ss_*
Successfully installed: /var/lib/ecss/ss/ss_dnd.xml
Successfully installed: /var/lib/ecss/ss/ss_clir.xml
. . .
Successfully installed: /var/lib/ecss/ss/ss_chunt.xml
```

8.10 Первоначальная конфигурация системы с помощью web-конфигуратора

пример первоначальной конфигурации через Web интерефейс

8.10.1 Исходные данные

- Установка системы завершена;
- Система готова к дальнейшей настройке;
- Интерфейсы запущены.

Рекомендуется использовать последние доступные версии браузеров. Рекомендуемые к использованию браузеры: Яндекс, Chrome.

Для начала конфигурации системы необходимо зайти в web-конфигуратор по адресу http://<Ipadd SSW>.

Для определения и регистрации в системе планируются:

- Абоненты с номерами 42001, 42002 , 42003, 42004 ,42005, 42006, 42007, 42008, 42009, 42010;
- Транк в сторону шлюза.

8.10.2 Подготовка к работе

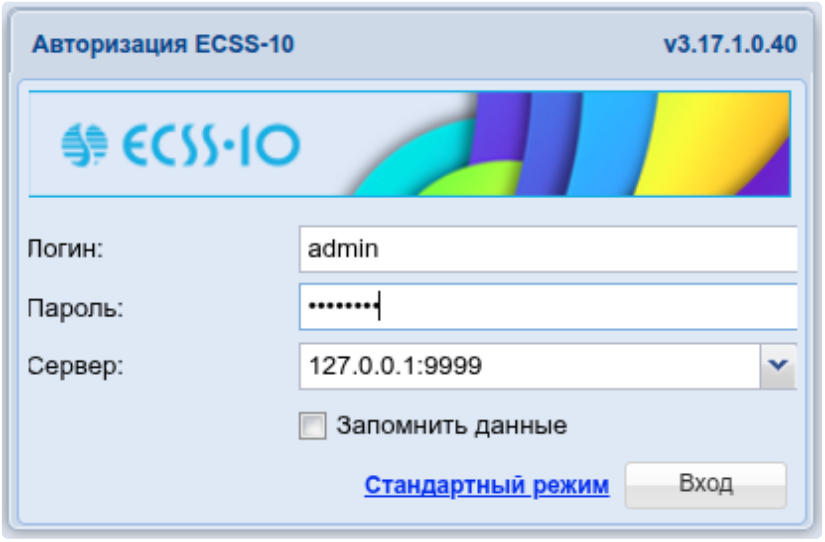


Рисунок 3 — Вход в web-конфигуратор (окно авторизации)

В окне авторизации необходимо ввести значения, определенные при установке web-конфигуратора.

> ⚠ Стандартные значения для авторизации:
> **Логин:** admin
> **Пароль:** password

После входа в систему будет видна основная рабочая область с иконками приложений, а также строка состояния с доступными опциями, в частности:

- 1 — выход из системы;
- 2 — выбор домена;
- 3 — выбор языка.

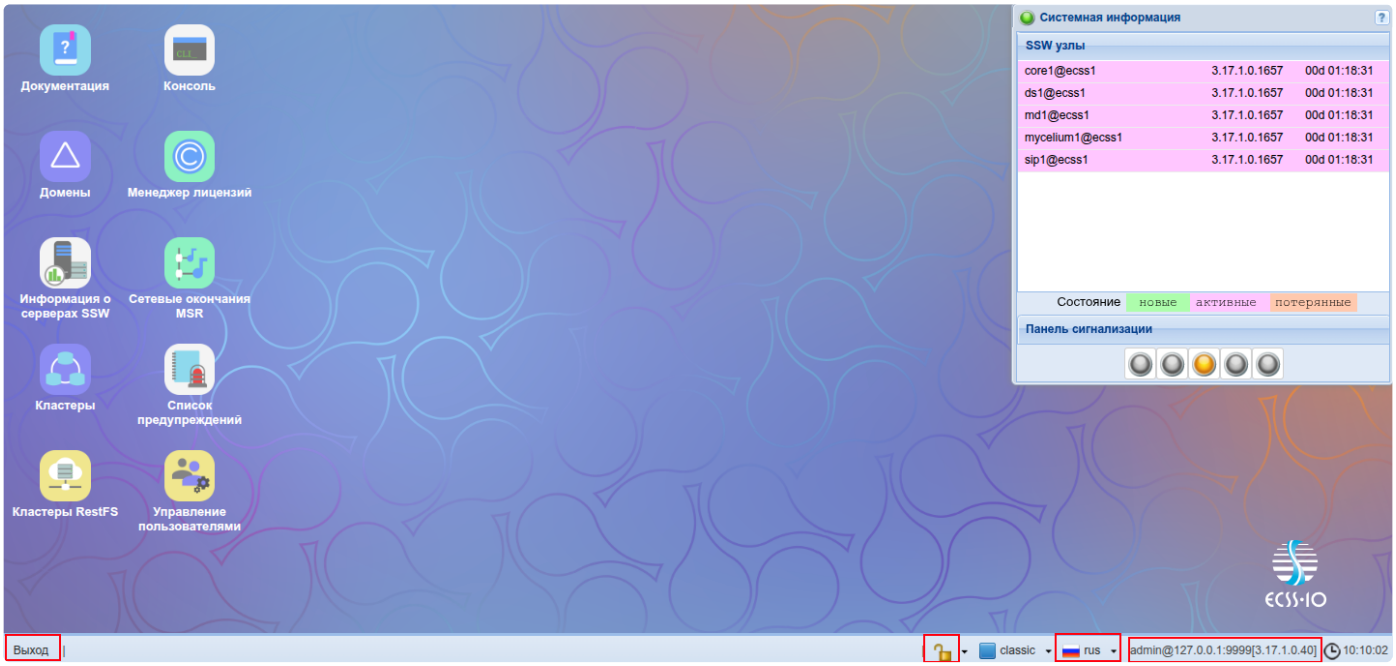


Рисунок 4 — Вид рабочей области web-конфигуратора

8.10.3 Создание учетной записи оператора

После авторизации для повышения безопасности при эксплуатации программного коммутатора рекомендуется создание учетных записей для операторов, а также сменить пароль для пользователя **admin**.

Для создания новой учетной записи оператора используется приложение "Управление пользователями" ("User manager"):

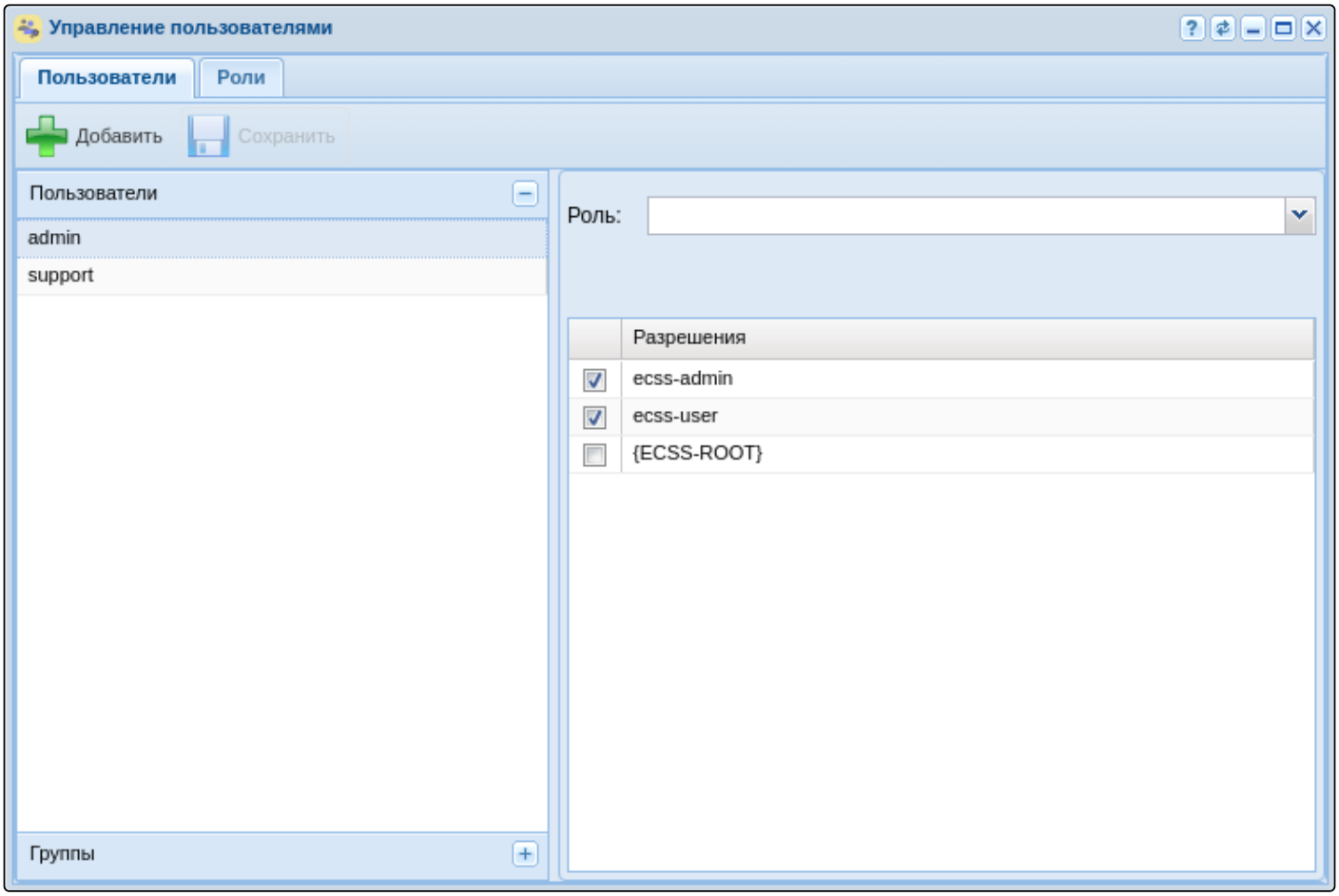


Рисунок 5 – Вид приложения "*Управление пользователями*"

Нажмите кнопку "*Добавить*"  . В открывшемся окне требуется определить новую учетную запись, для этого:

1. В поле "*Имя*" введите логин учетной записи, например "test";
2. В поле "*Пароль*" и "*Подтверждение*" введите пароль для пользователя, например "testpassword";
3. Определите уровень прав доступа для пользователя через выбор текущих разрешений или с помощью ролей, например **ecss-user**. Описание можно посмотреть в разделе "Управление пользователями".

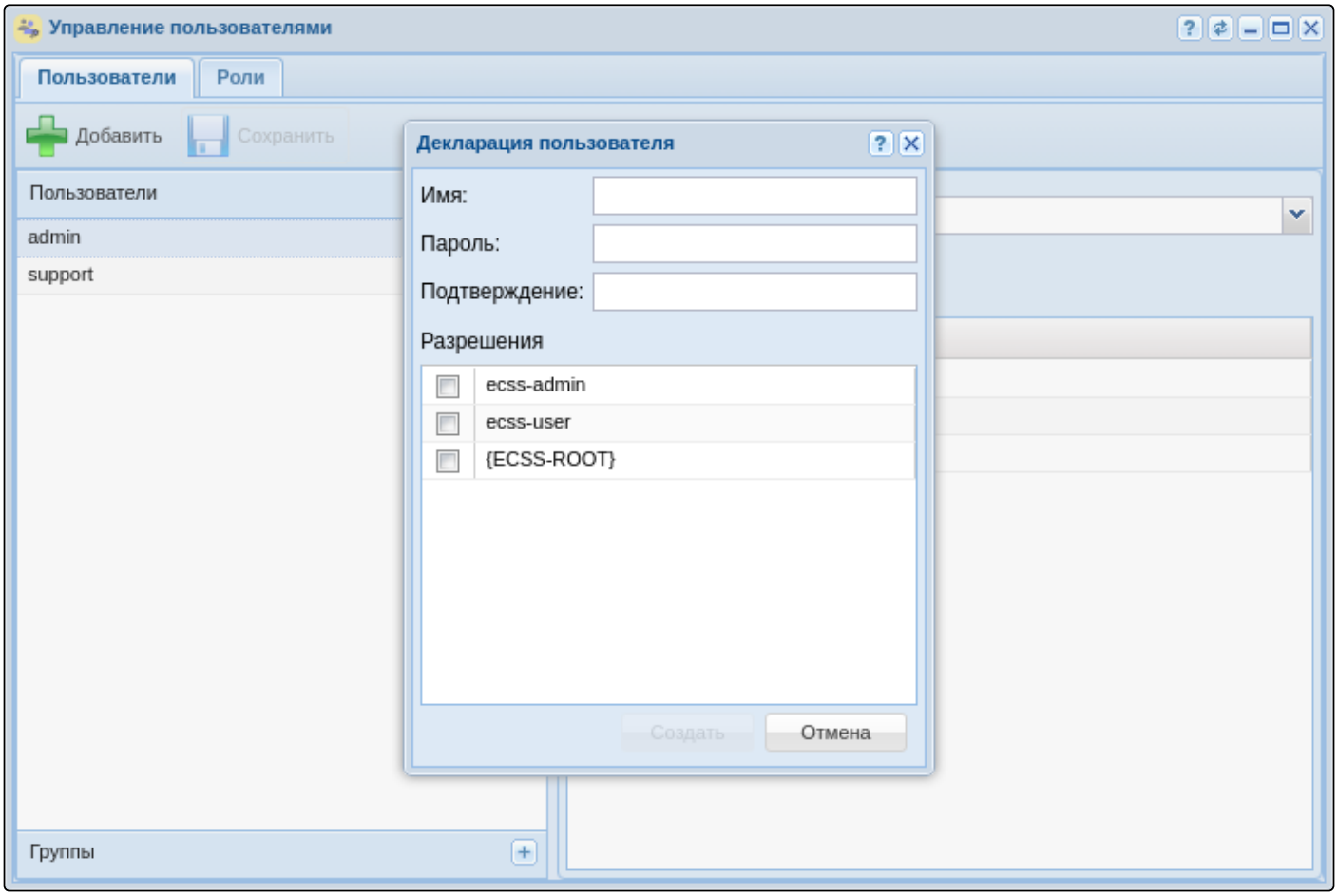


Рисунок 6 – Диалоговое окно создания учетной записи оператора

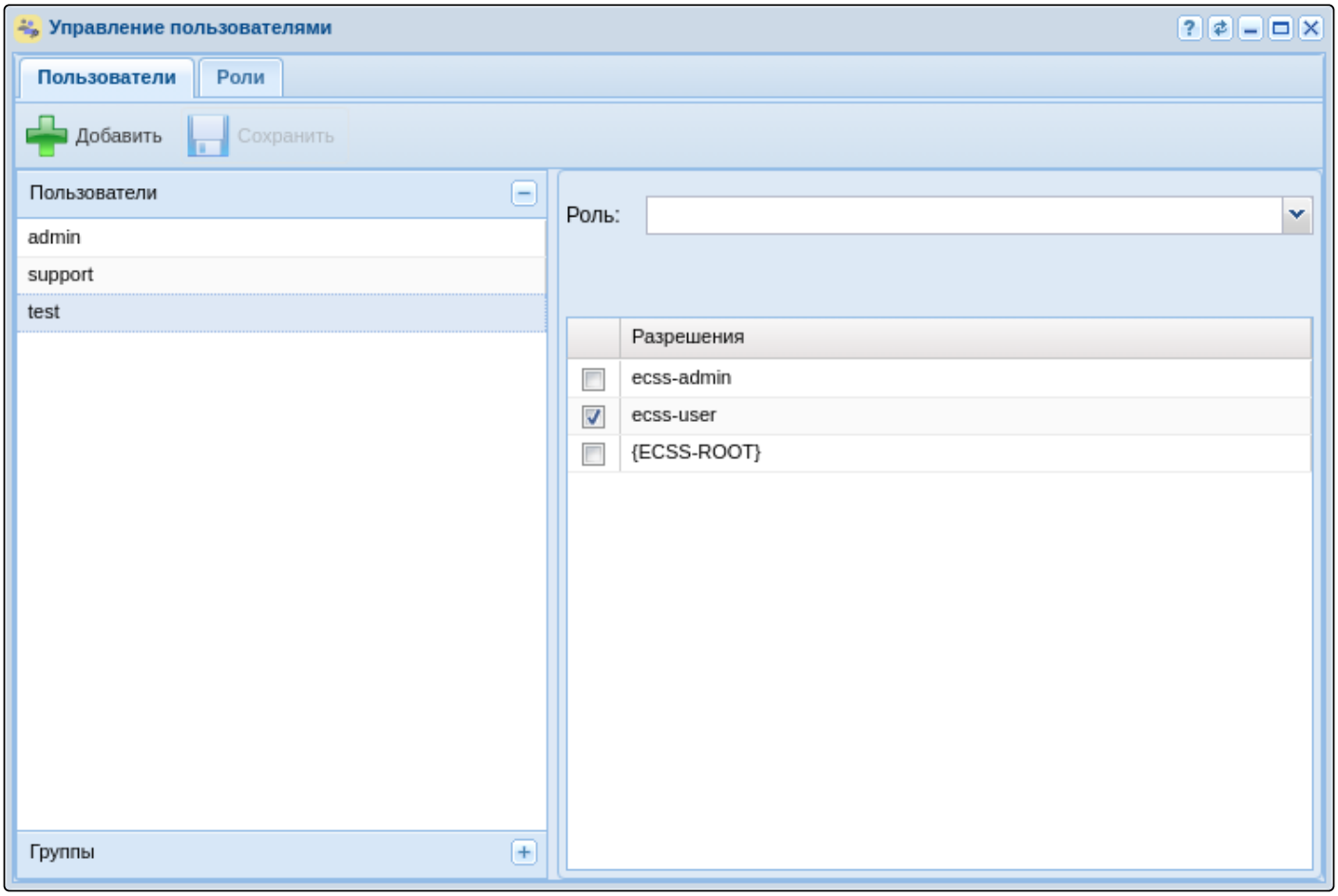


Рисунок 7 — Вид приложения с созданной учетной записью оператора

Для смены пароля рядом с названием пользователя нажмите на кнопку редактировать. В появившемся диалоговом окне введите:

1. Старый пароль (для пользователя "**admin**" по умолчанию паролем является "**password**");
2. Новый пароль;
3. Подтвердите новый пароль.

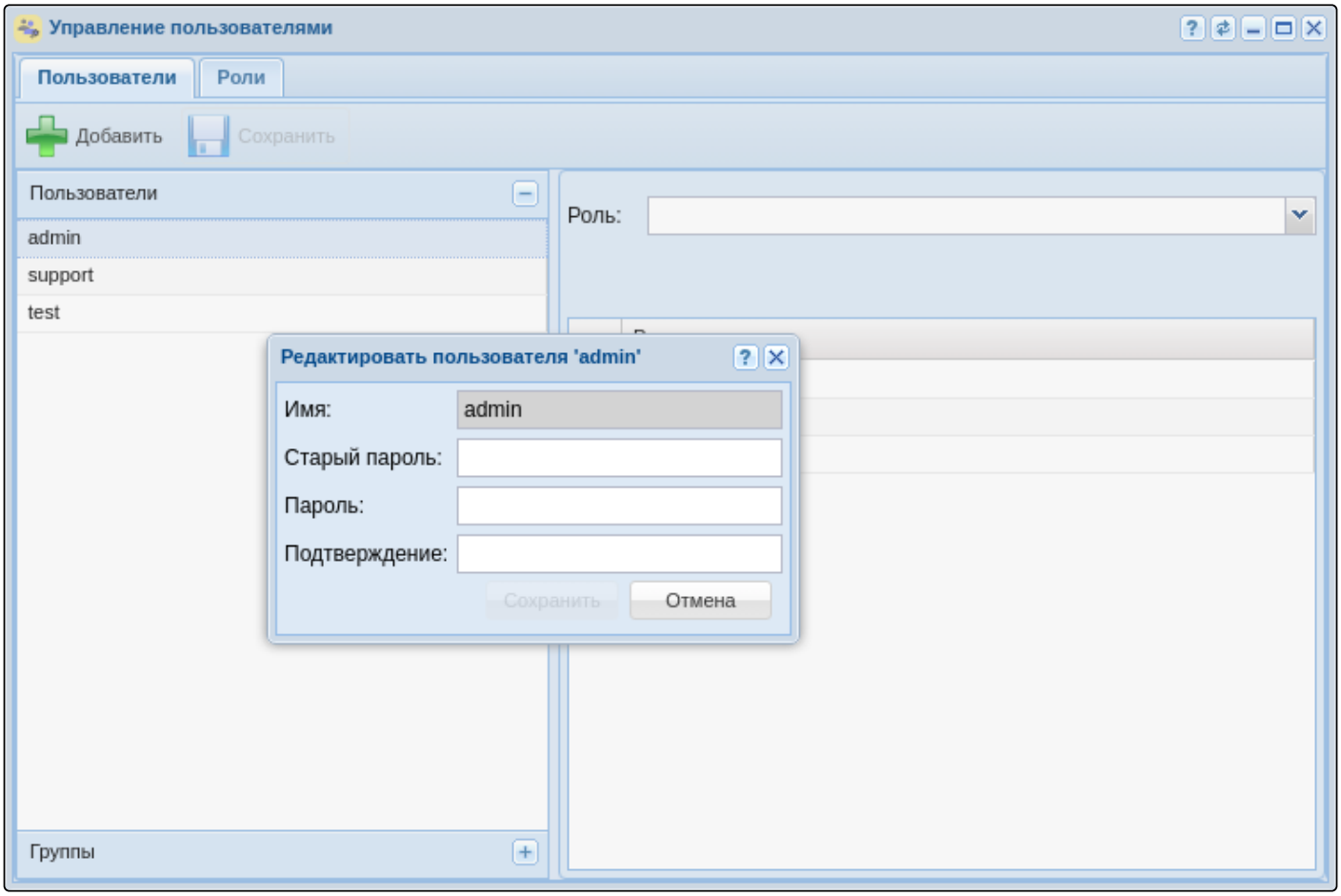
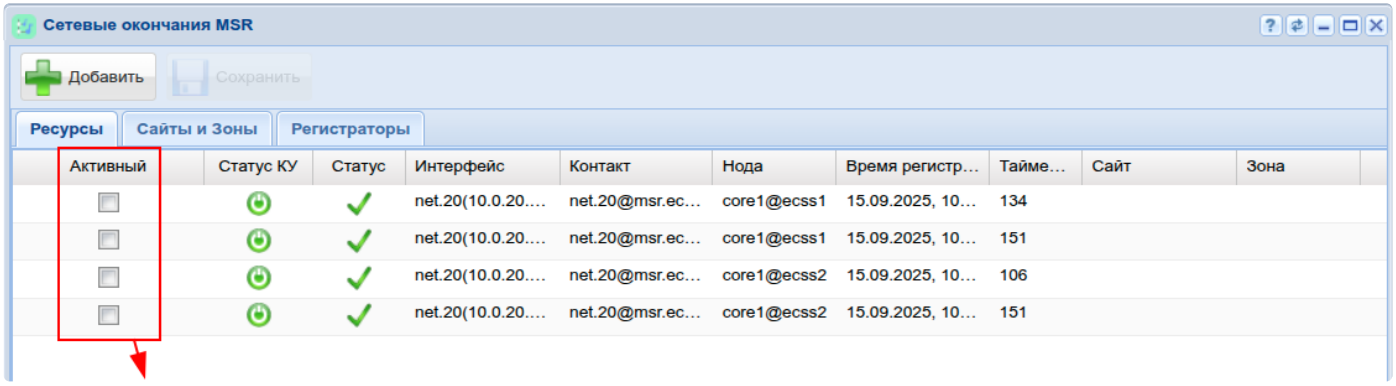


Рисунок 8 — Диалоговое окно редактирования учетной записи

8.10.4 Добавление медиа-ресурсов.

Для того чтобы начать работу с медиа-сервером сразу после его регистрации, необходимо активировать сетевые интерфейсы MSR. Приложение web-конфигуратора "Сетевые окончания MSR" ("MSR registrars") служит для настройки медиа-ресурсов.



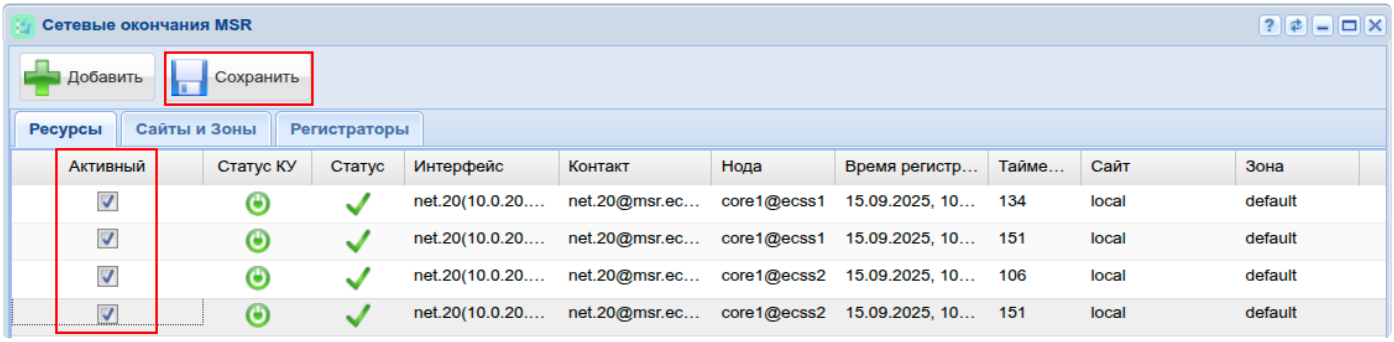
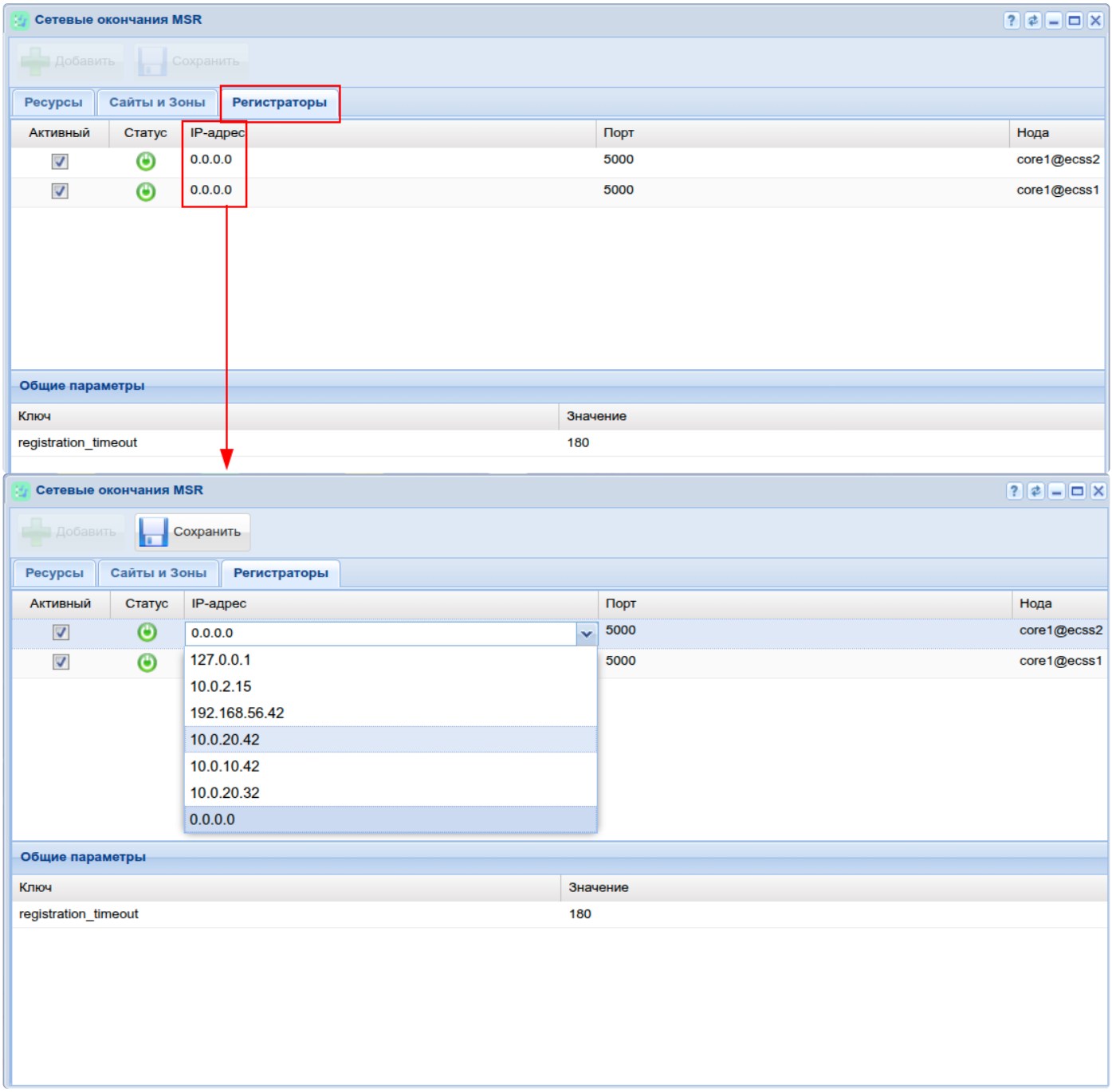


Рисунок 9 — Сетевые окончания MSR

Установить ipадрес для приема соединений от медиасервера, изменив значение 0.0.0.0 на реальные адреса MSR:



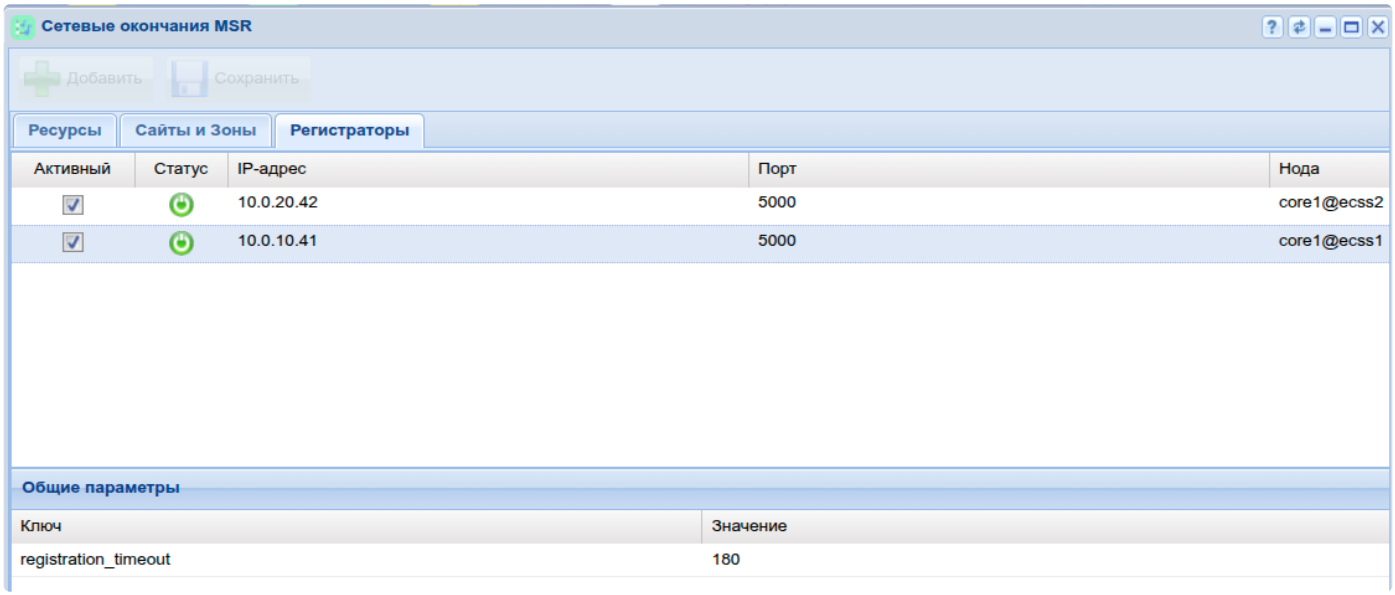


Рисунок 10 — Установка IP-add регистратора MSR

8.10.5 Создание IP-set (sip-транспорта) и присвоение его к домену

IP-set — совокупность одного или нескольких IP-адресов из одной подсети и списка слушающих UDP/TCP-портов, которые будут открыты SIP-адаптером на данных адресах.

Для настройки интерфейса необходимо открыть приложение "Кластеры ("Clusters").

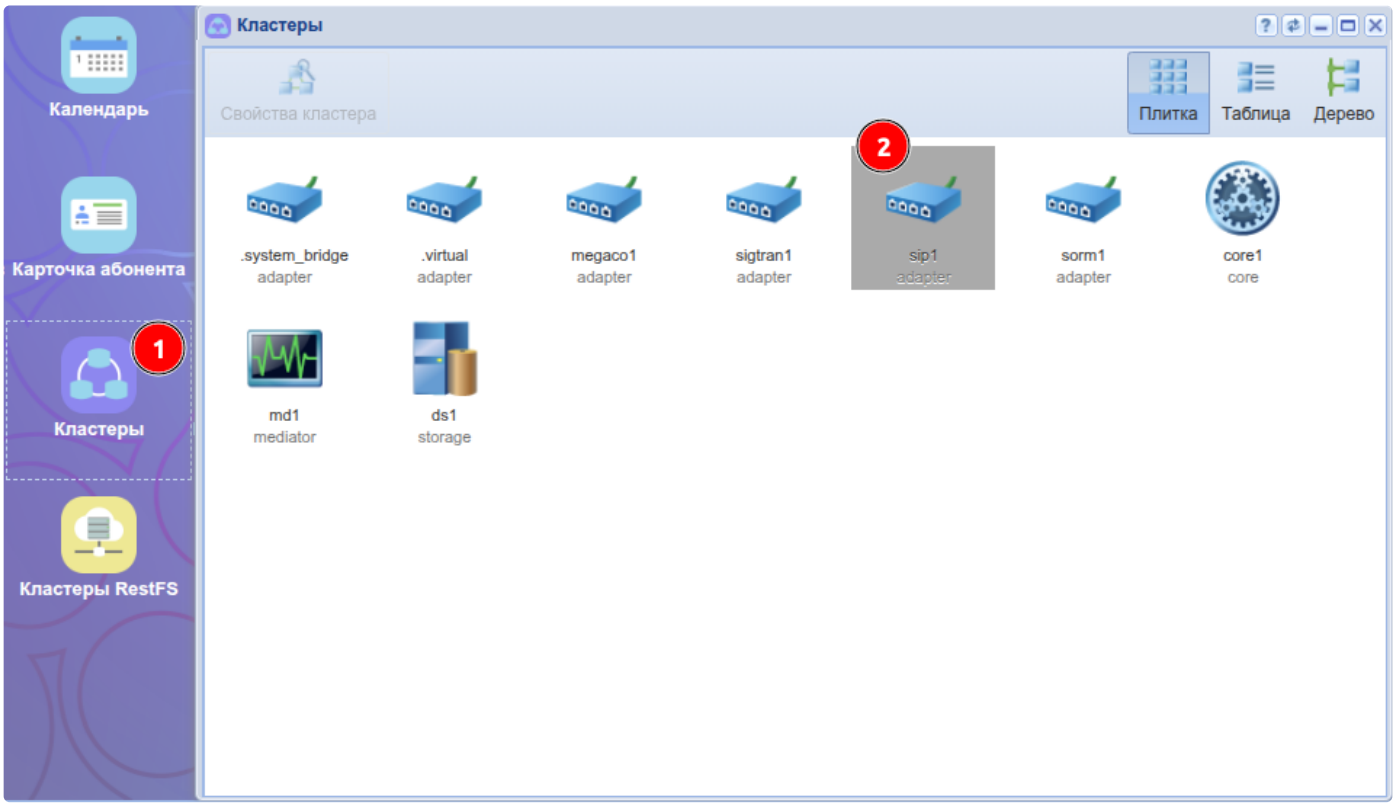


Рисунок 11 — Вид приложения "Кластеры"

Для создания новой группы IP-адресов (IP-set) выделите кластер SIP-адаптера "sip1" и нажмите на кнопку *"Свойства кластера"* (или нажмите на иконку кластера двойным щелчком левой кнопки мыши).

В появившемся диалоговом окне перейдите на вкладку *"Транспорт"*. Далее нажмите кнопку *"Добавить"*. Появится новая группа, для редактирования полей двойным щелчком кнопки мыши нажмите на нужное:

1. Переименуйте группу адресов (IP-set), например "test_set";

2. Задайте порт, по которому будет происходить обращение к домену, например 5060;

3. Разверните вновь созданную группу нажатием на треугольник слева от названия группы;

4. Определите адрес для ноды SIP-адаптера, согласно примеру конфигурации. Для системы с резервированием укажите 10.0.20.31 и 10.0.20.32:

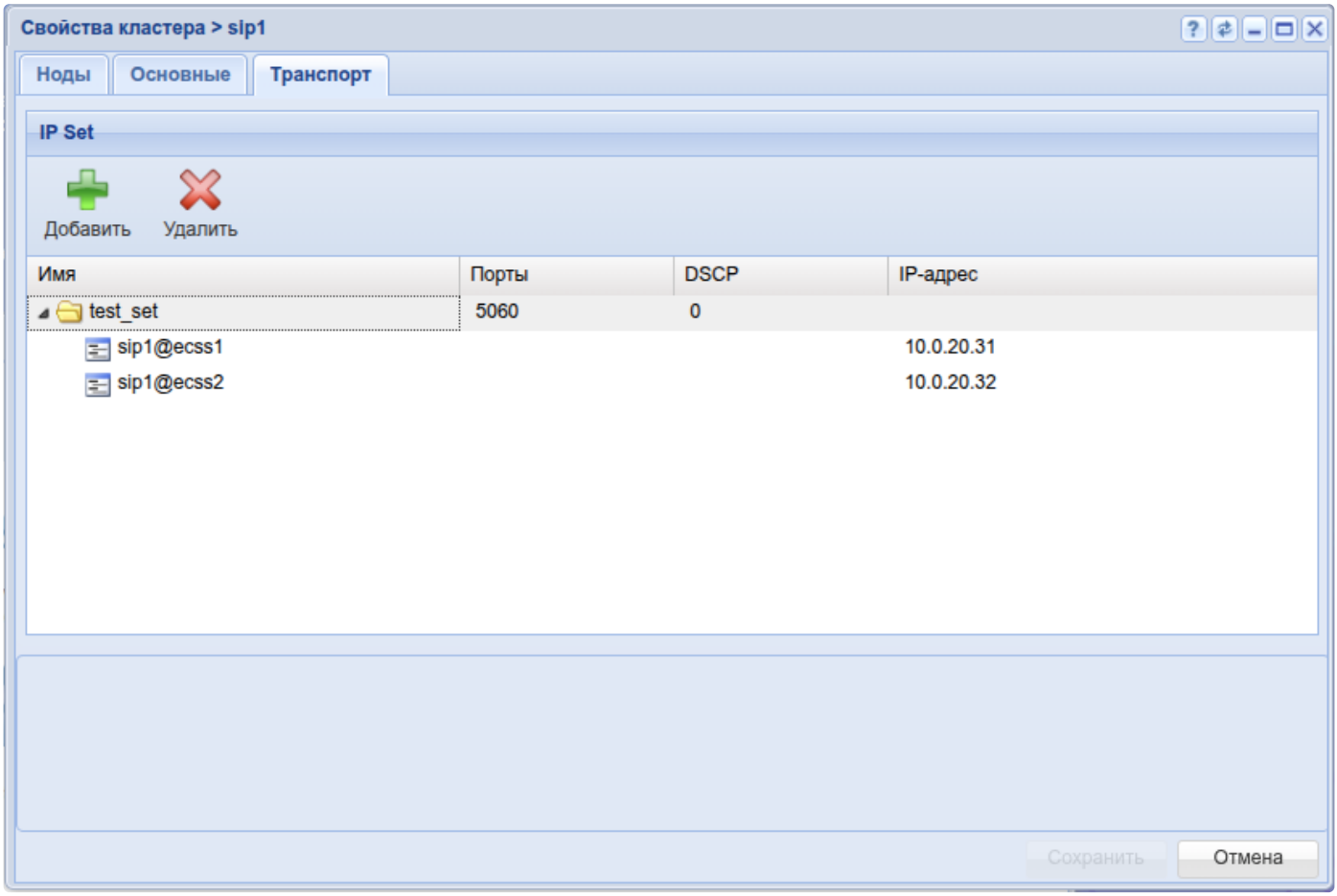


Рисунок 12 — Вид приложения "Кластеры/Транспорт" для кластера

Нажмите кнопку *"Сохранить"* для применения настроек.

8.10.6 Создание домена

Домен - Виртуальная PBX внутри нашего SSW. Таких виртуальных PBX может быть создано необходимое для проекта кол-во.

Для создания домена необходимо войти в приложение *"Домены" ("Domain")*. В открывшемся окне требуется создать домен, для этого:

1. Нажмите кнопку "*Добавить домен*":

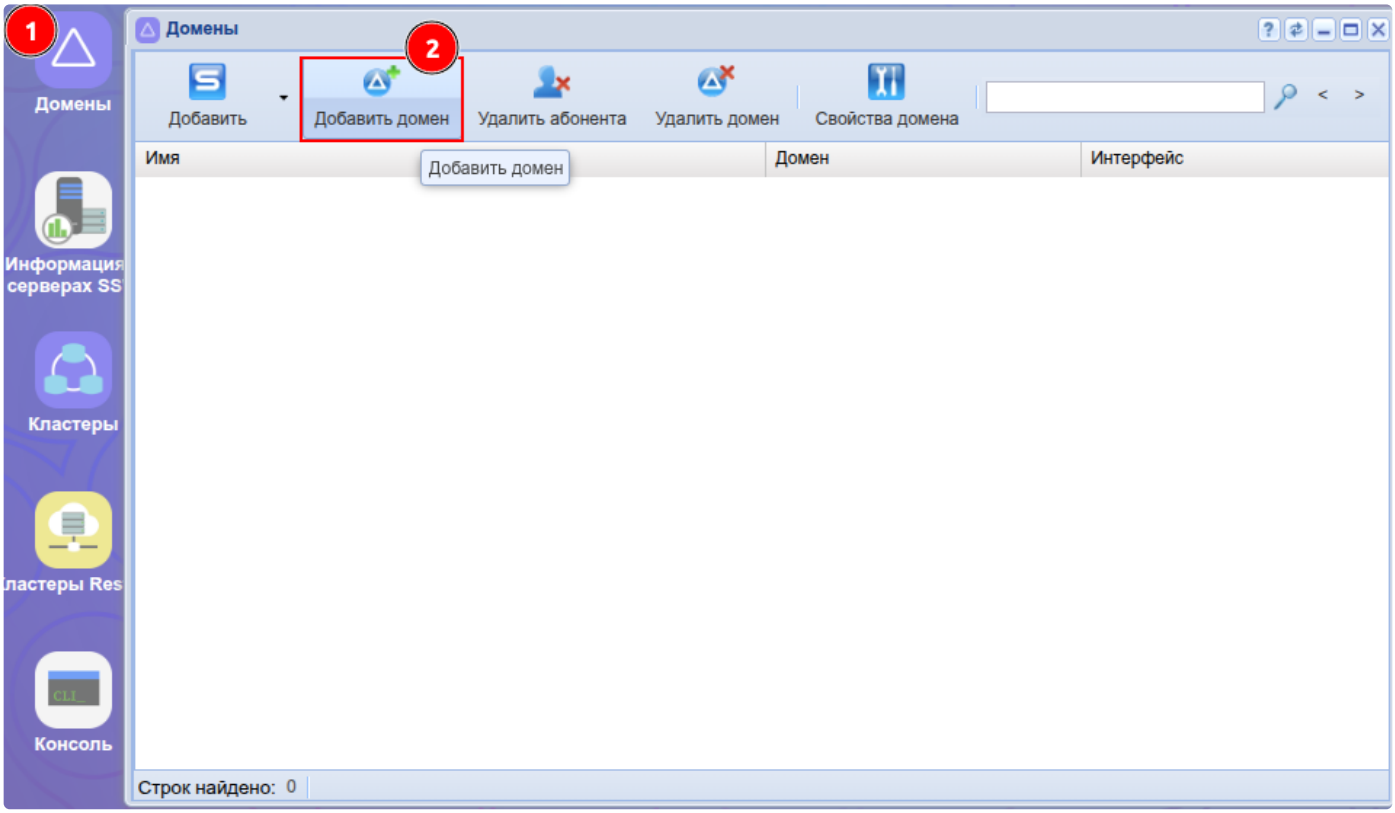


Рисунок 13 — Добавление домена в систему

2. В появившемся диалоговом окне доступны следующие настройки:

- *Имя (Name)* — индивидуальное имя виртуальной АТС;
- *Профиль услуг (SS profile)* — системный профиль дополнительных услуг. Этот профиль будет скопирован с тем же именем во вновь создаваемый домен и для всех услуг из этого профиля будет автоматически разрешён доступ через access-list;
- *IVR профиль (IVR profile)* — профиль IVR, заданный в приложении "Редактор IVR ограничений".

Введите название домена, например "test_domain";

3. Нажмите кнопку "*Ок*":

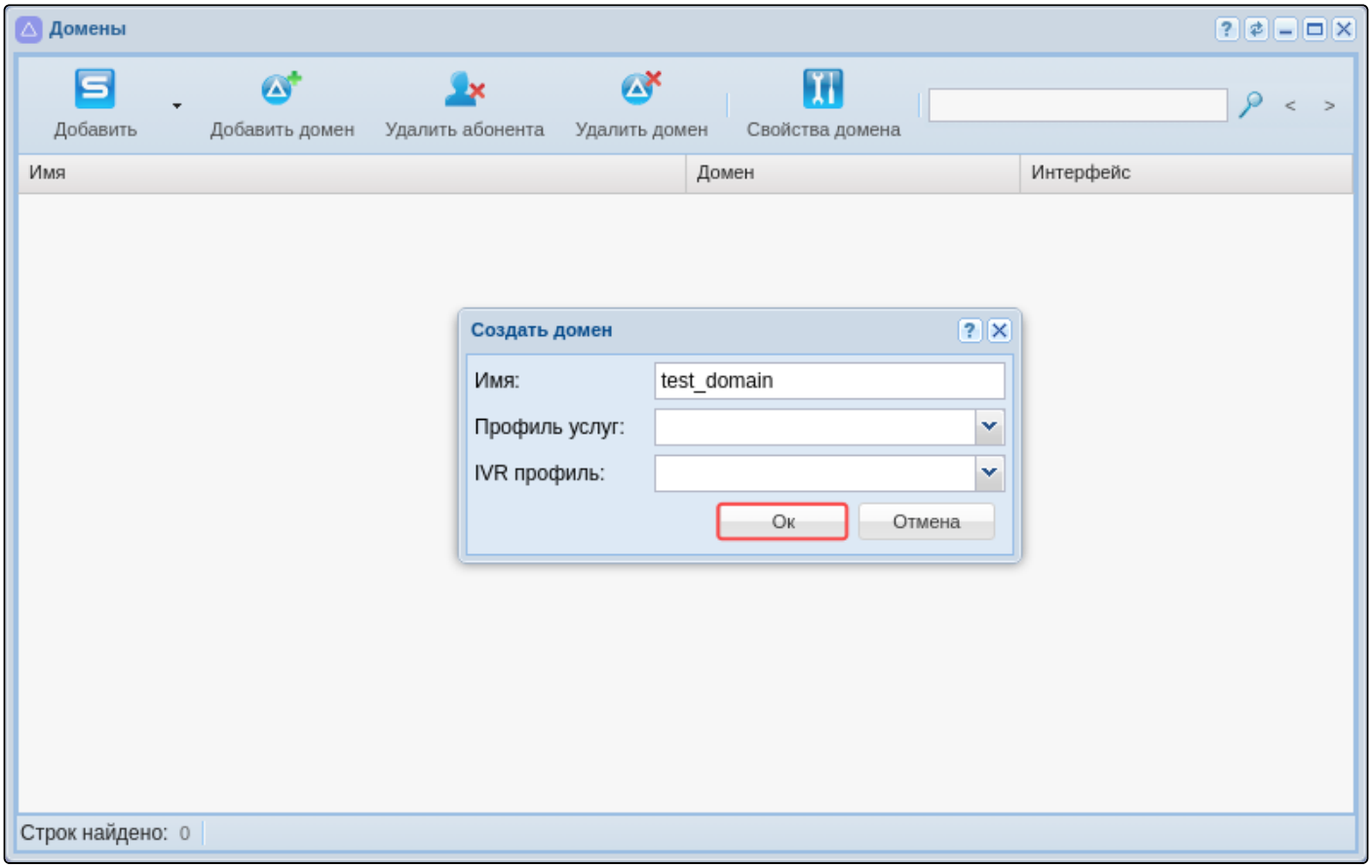


Рисунок 14 – Определение настроек нового домена

4. Нажмите кнопку *"Обновить"* в приложении.

Созданный домен будет отображаться в текущей конфигурации:

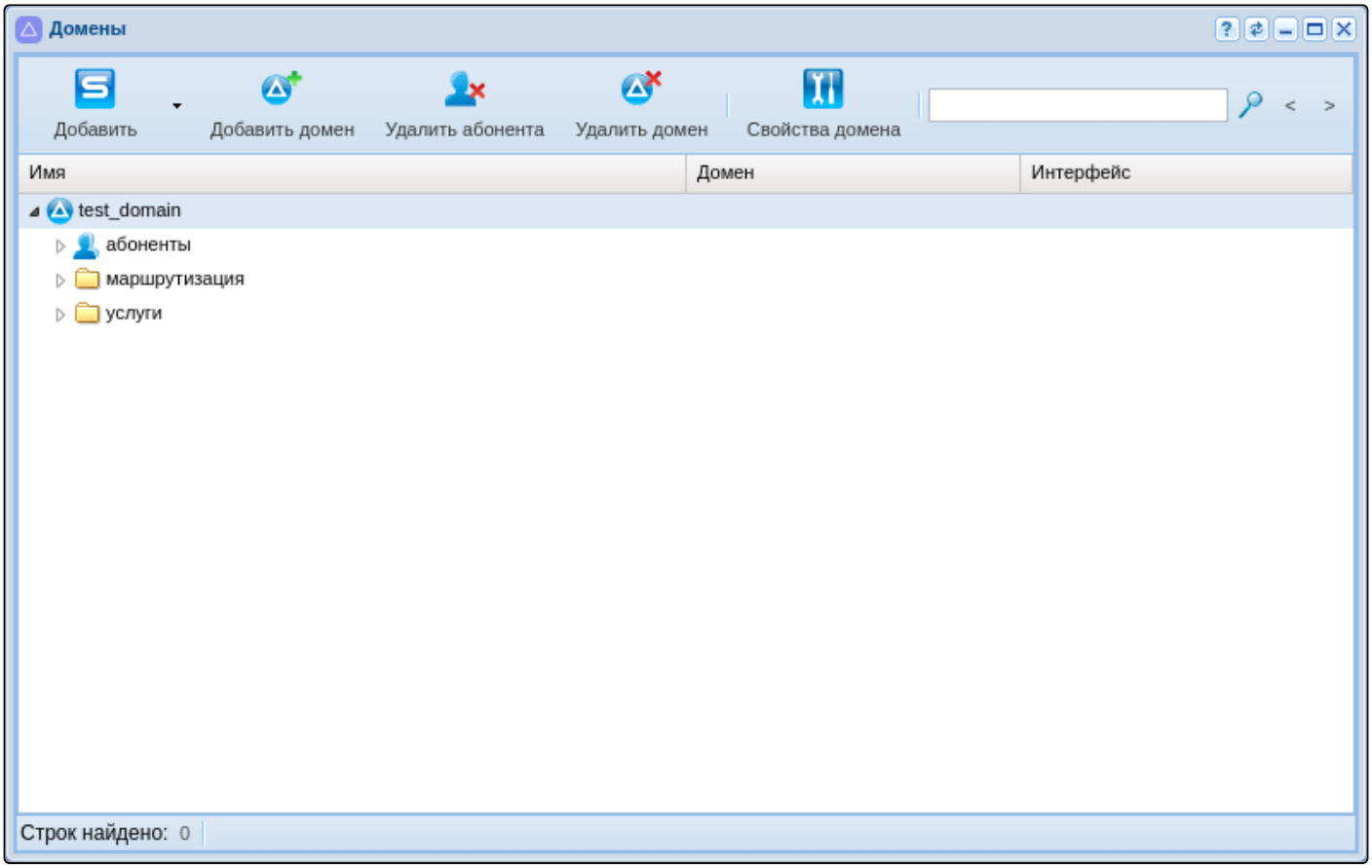


Рисунок 15 — Отображение созданного домена

Для редактирования текущего домена его необходимо выбрать в системе. Для перехода в домен воспользуйтесь опцией выбора домена (см. пункт 2 на рисунке "Вид рабочей области web-конфигуратора").

После выбора домена, согласно текущей конфигурации системы, станут доступны все приложения:

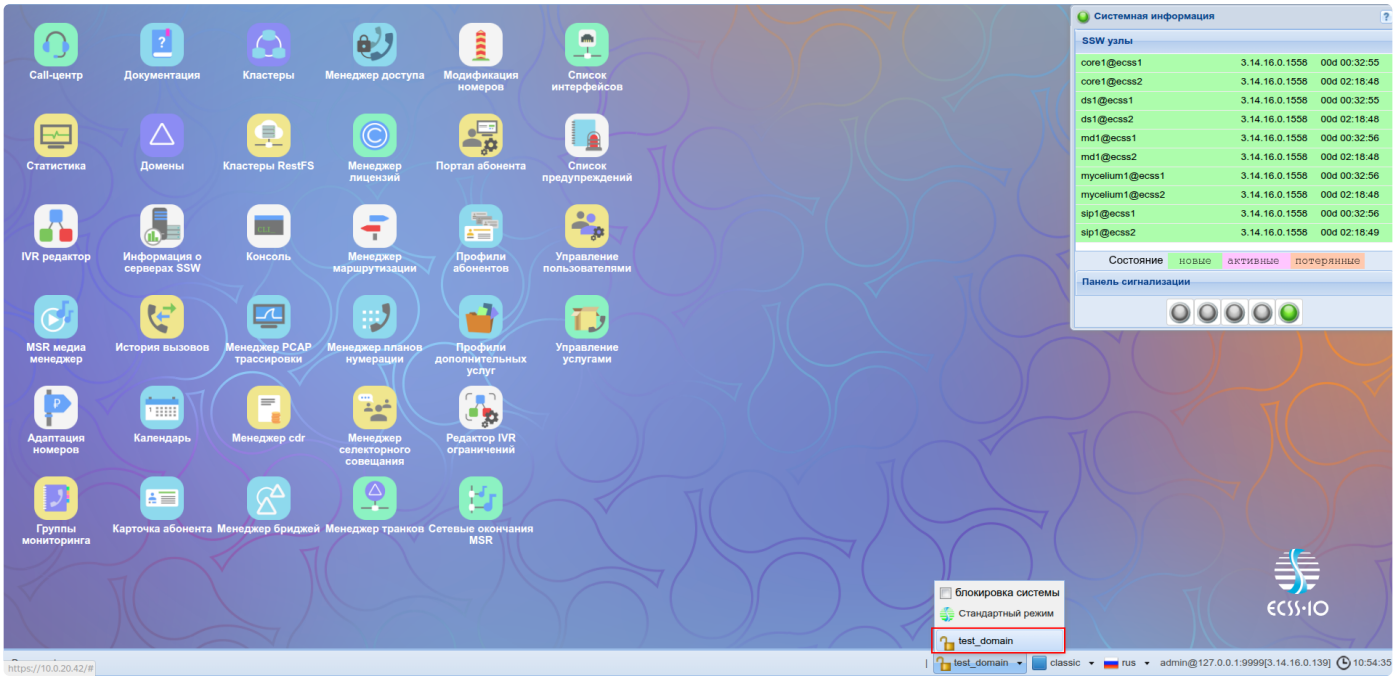


Рисунок 17 — Отображение приложений в текущей конфигурации системы

Для того чтобы **привязать группу адресов к домену**, необходимо вернуться в приложение *"Домены" ("Domain")*, выделить домен и перейти в настройки нажатием на кнопку *"Свойства домена"* или двойным нажатием левой кнопки мыши на домен.

В настройках списка откройте ветку "SIP", в ней "SIP транспорт", далее выберите в поле IP set созданную группу адресов. Нажмите кнопку *"Сохранить"* для применения настроек.

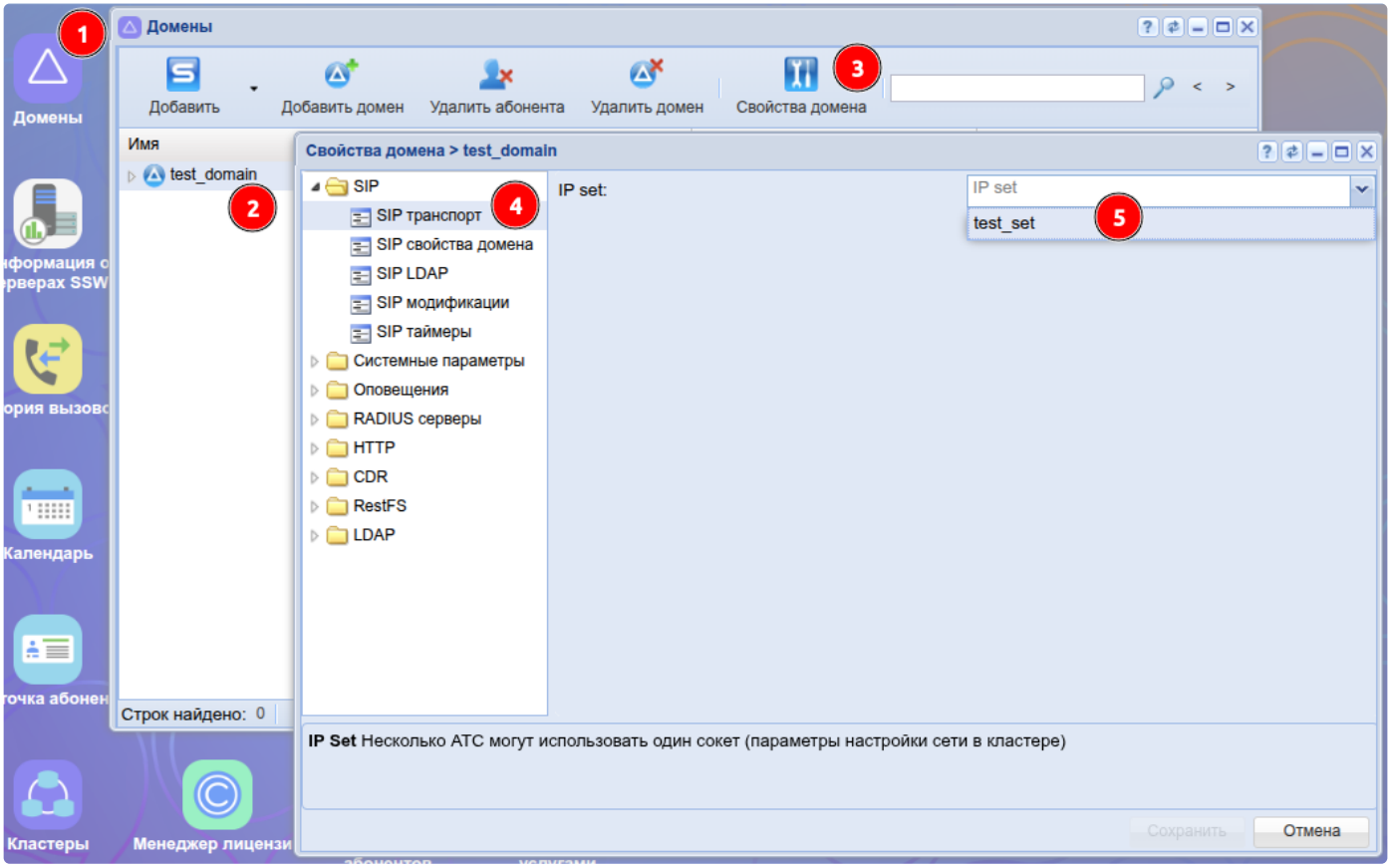


Рисунок 18 — Окно настройки для SIP транспорта

Выбрать требуемые услуги для созданного домена, которые будут использоваться (для примера для абонентского домена транковые услуги не востребованы , и наоборот для транзитного домена не нужно указывать услуги для абонента).

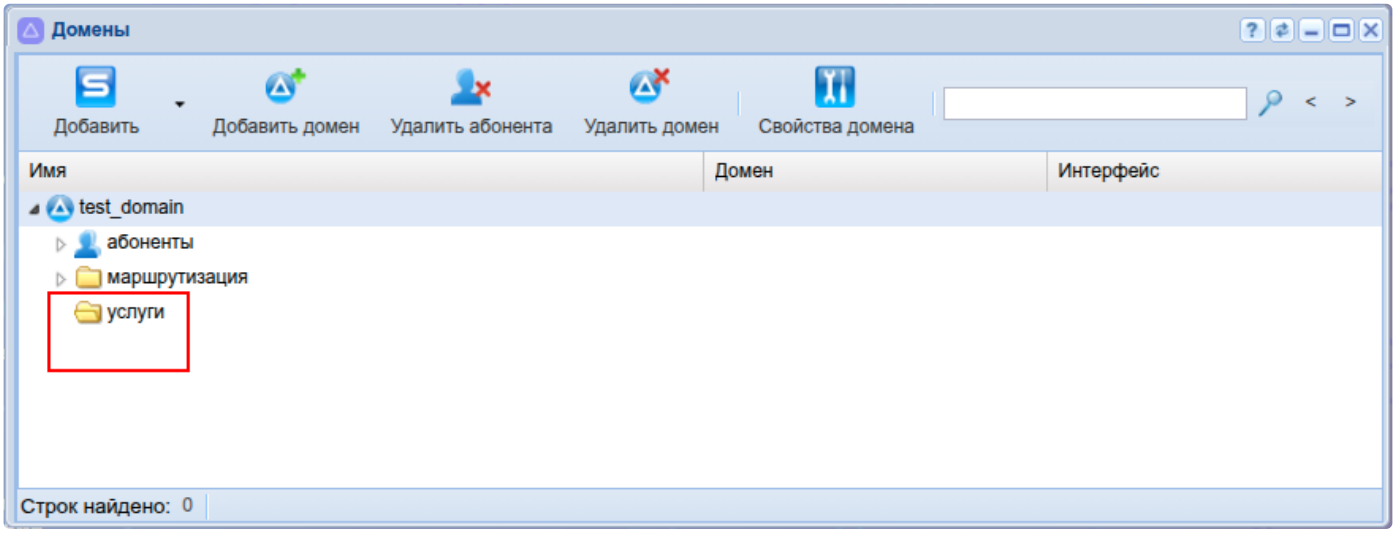


Рисунок 19 — После создания нового домена, услуги отсутствуют.

1. Выбрать приложение "Управление услугами"
2. Выбрать колонку "Установленные услуги" (услуги указанные в лицензии для SSW)

3. Выбрать необходимые для указанного домена услуги
4. Сохранить конфигурацию

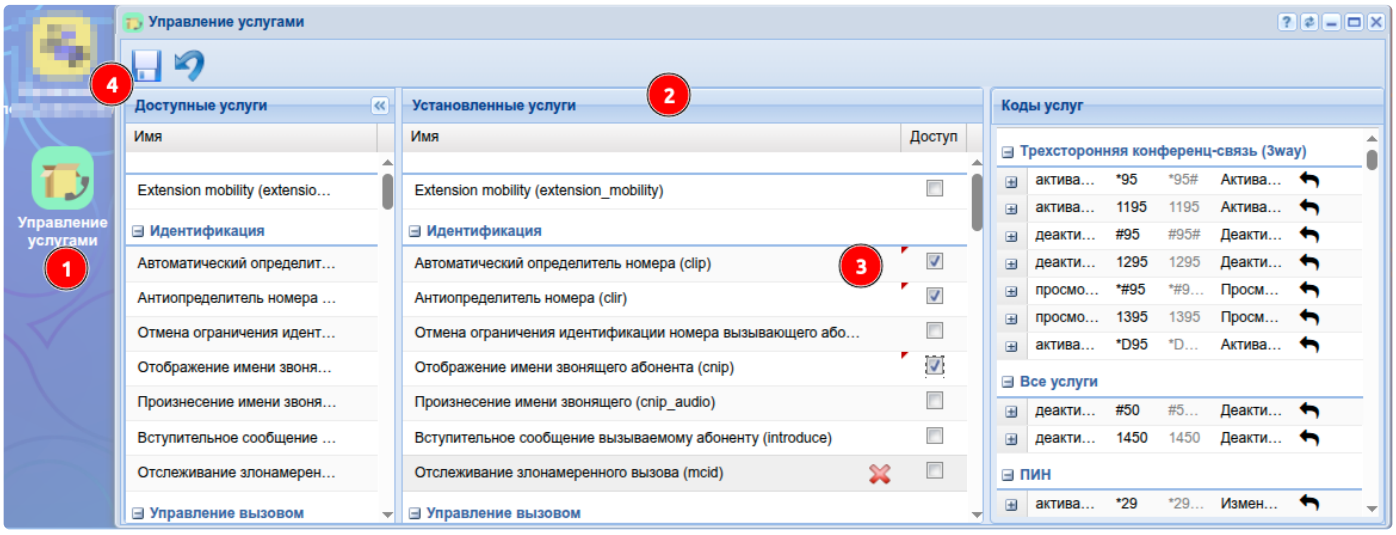


Рисунок 20 — Выбор услуг для абонентов / транков нового домена.

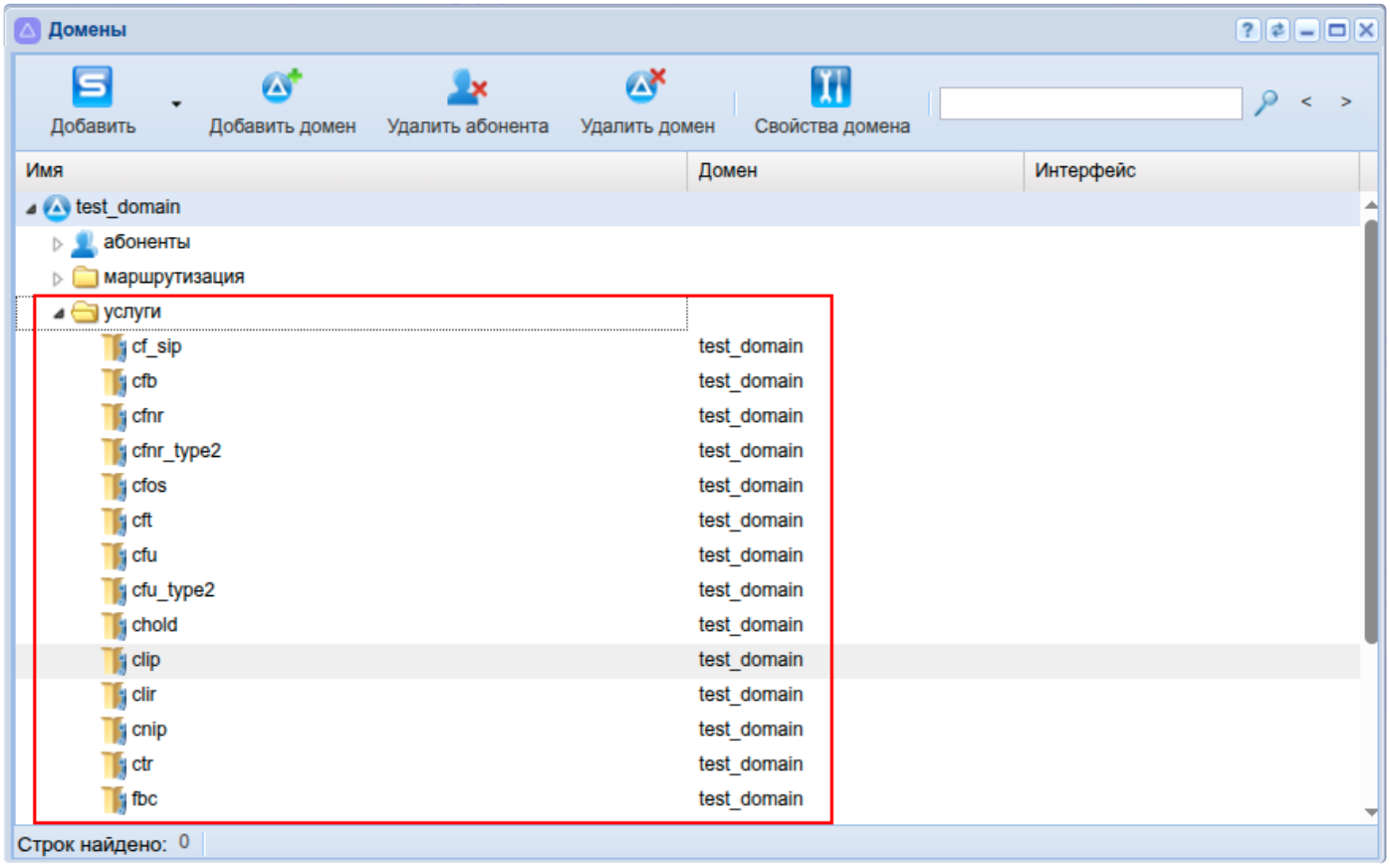


Рисунок 21 — Список услуг теперь доступных для абонентов нового домена.

8.10.7 Создание абонентов

Для создания и редактирования параметров абонентов в системе используется приложение "Карточка абонента" ("Subscriber card").

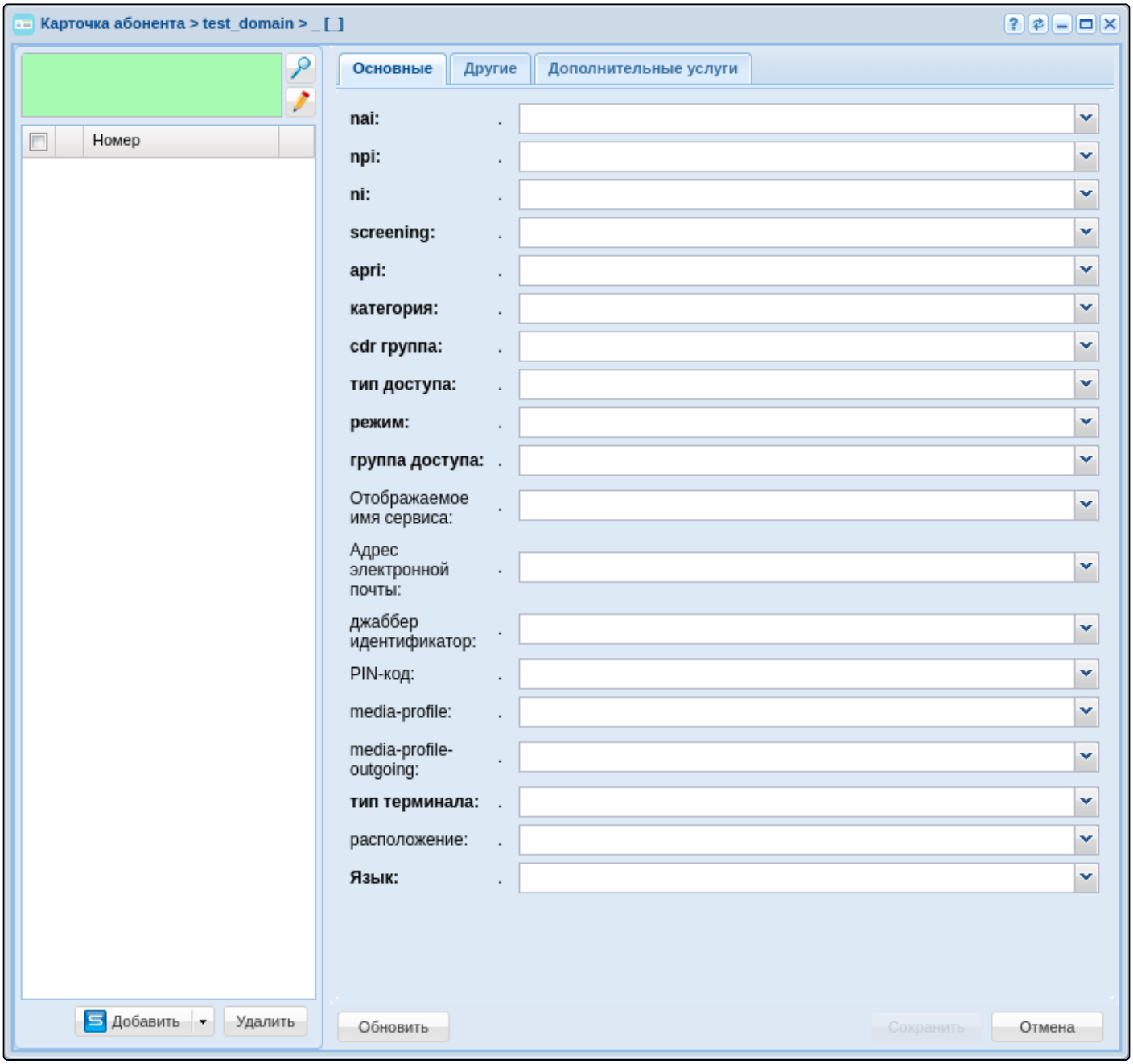


Рисунок 22 — Вид приложения "Карточка абонента"

В конфигурации возможно создание SIP-абонентов и виртуальных абонентов.

Для пользователей, имеющих физическое окончание, используется функционал SIP-абонента, в то время как виртуальный абонент используется тогда, когда нужна функциональность без физических окончаний. Например, номер для обращения к ivr-скрипту.

Для создания новых абонентов нужно нажать на кнопку *"Добавить"*.

В появившемся диалоговом окне укажите следующие параметры:

- *Имя интерфейса* — номер или группа номеров, который присваивается абоненту, строка, например {42001-42010} или abcd;
- *Контекст* — контекст маршрутизации, выбираем тот, который создали, например "test_name", или по умолчанию - "default_routing";
- Группа интерфейсов — имя группы объединяющей нескольких абонентов, для примера "sip";
- *Алиас как пользователь* — настройка, которая связывает номер сущности, алиас и пользователя одним и тем же названием, в примере настройка активирована;

- *Авторизация* — порядок проверки подлинности прав пользователя на доступ к данным, в примере используется"register";
- *Логин* — использую настройку номер как логин для простоты;
- *Пароль* — можно задать или использовать сгенерированный системой.

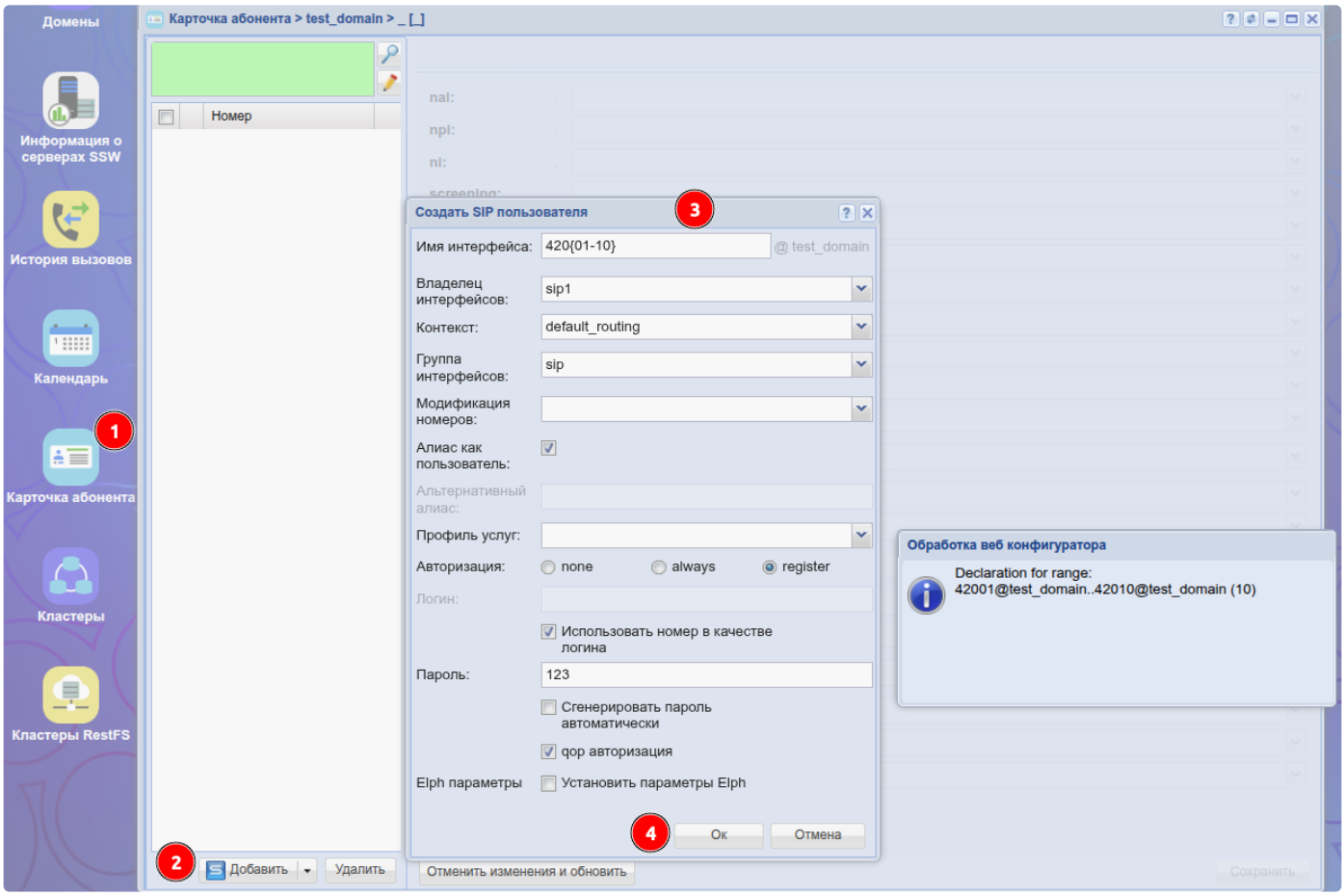


Рисунок 23 — Пример создания 10 абонентов в домене "test_domain"

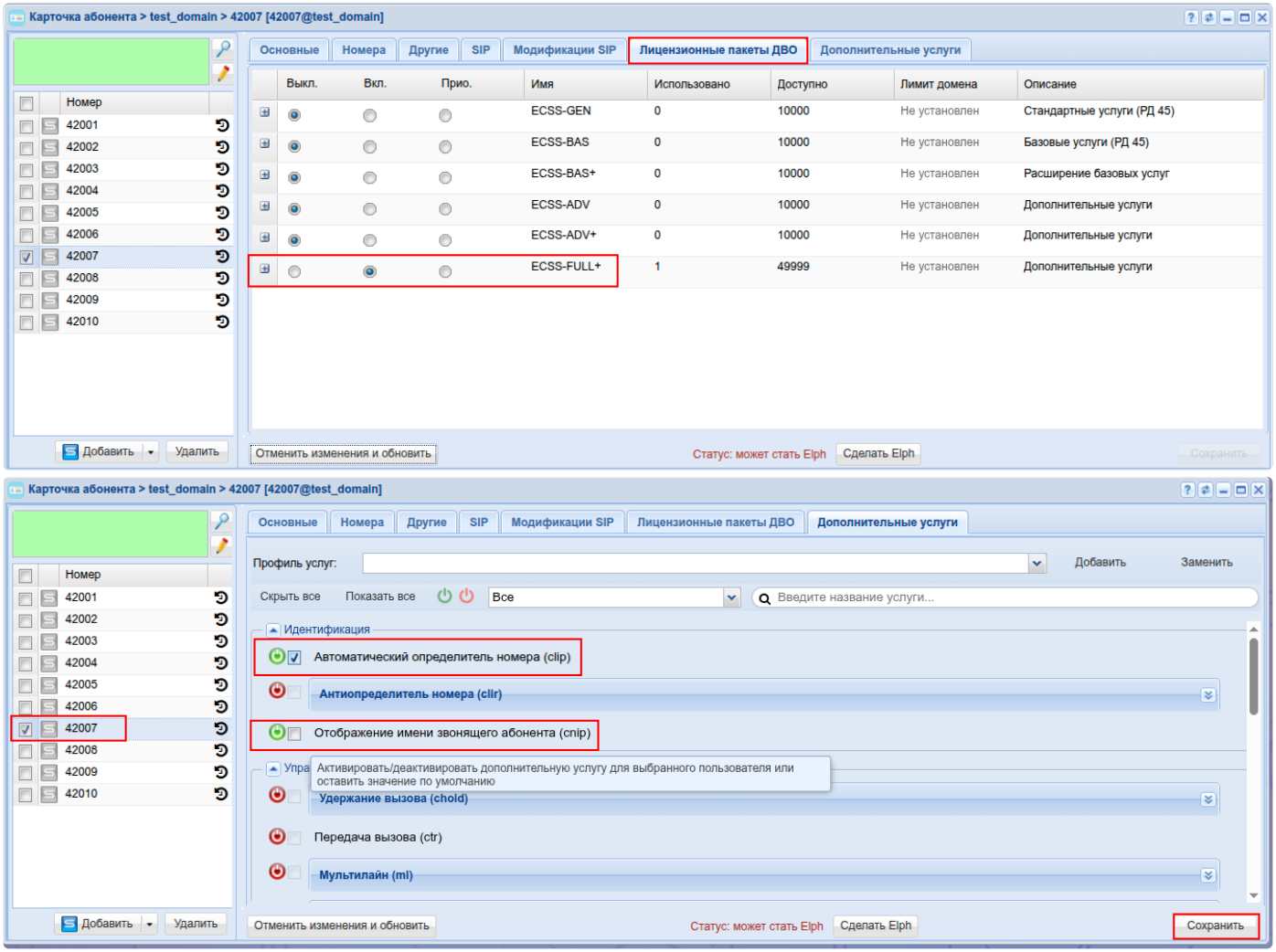


Рисунок 24 — Пример добавления услуг для абонента

8.10.8 Создание и применение контекстов маршрутизации для домена

Маршрутизация отвечает за поиск номера и последующую адресацию вызова. Для корректной работы системы должен быть настроен как минимум один контекст маршрутизации.

Настройка маршрутизации осуществляется в приложении "Менеджер маршрутизации" ("Routing manager") .

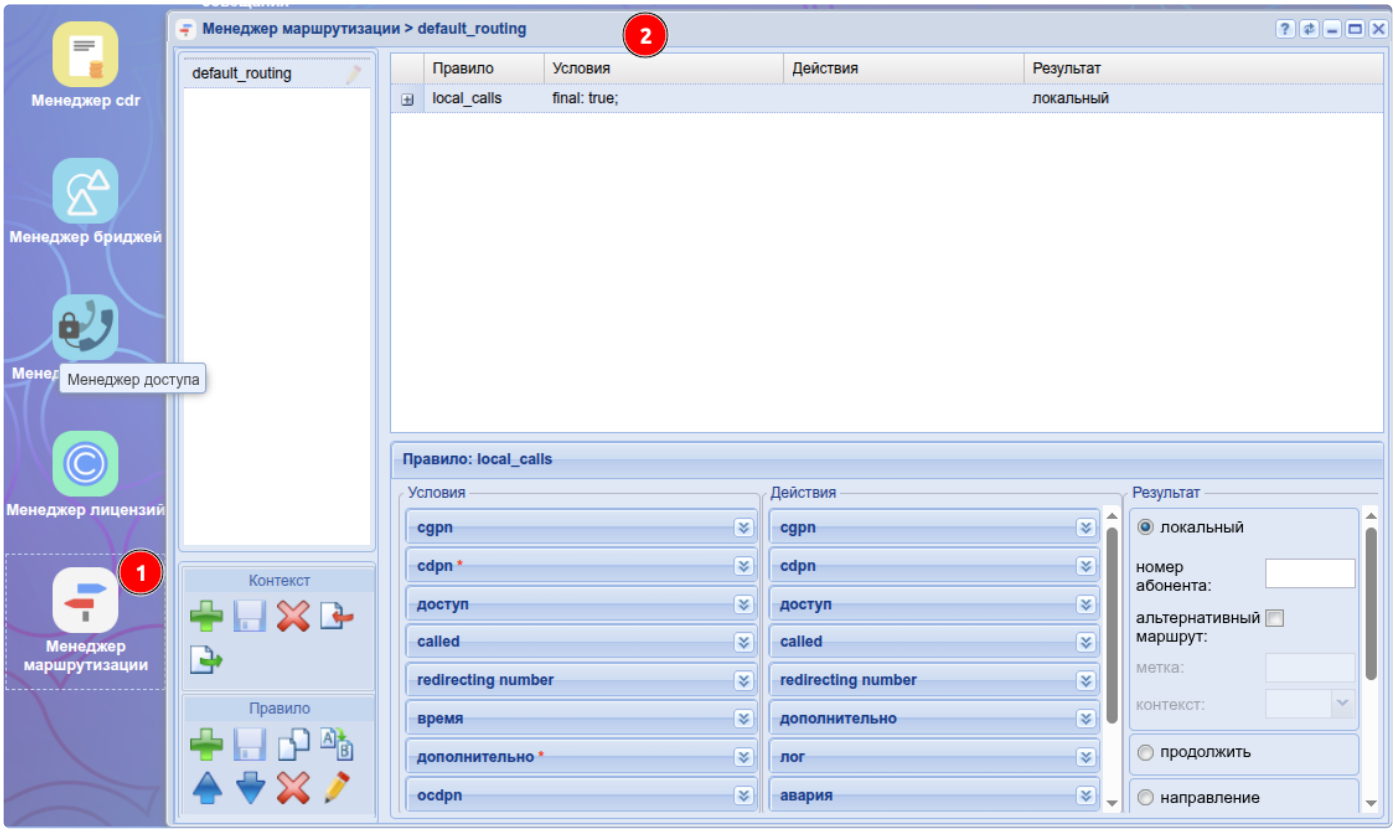


Рисунок 25 — Вид приложения "Менеджер маршрутизации"

В качестве примера создадим контекст, а затем несколько правил в нём, для этого:

1. В левой части окна в разделе "Контекст" нажмите на кнопку *"Создать контекст"*  ;
2. В открывшемся диалоговом окне обозначьте имя контекста:

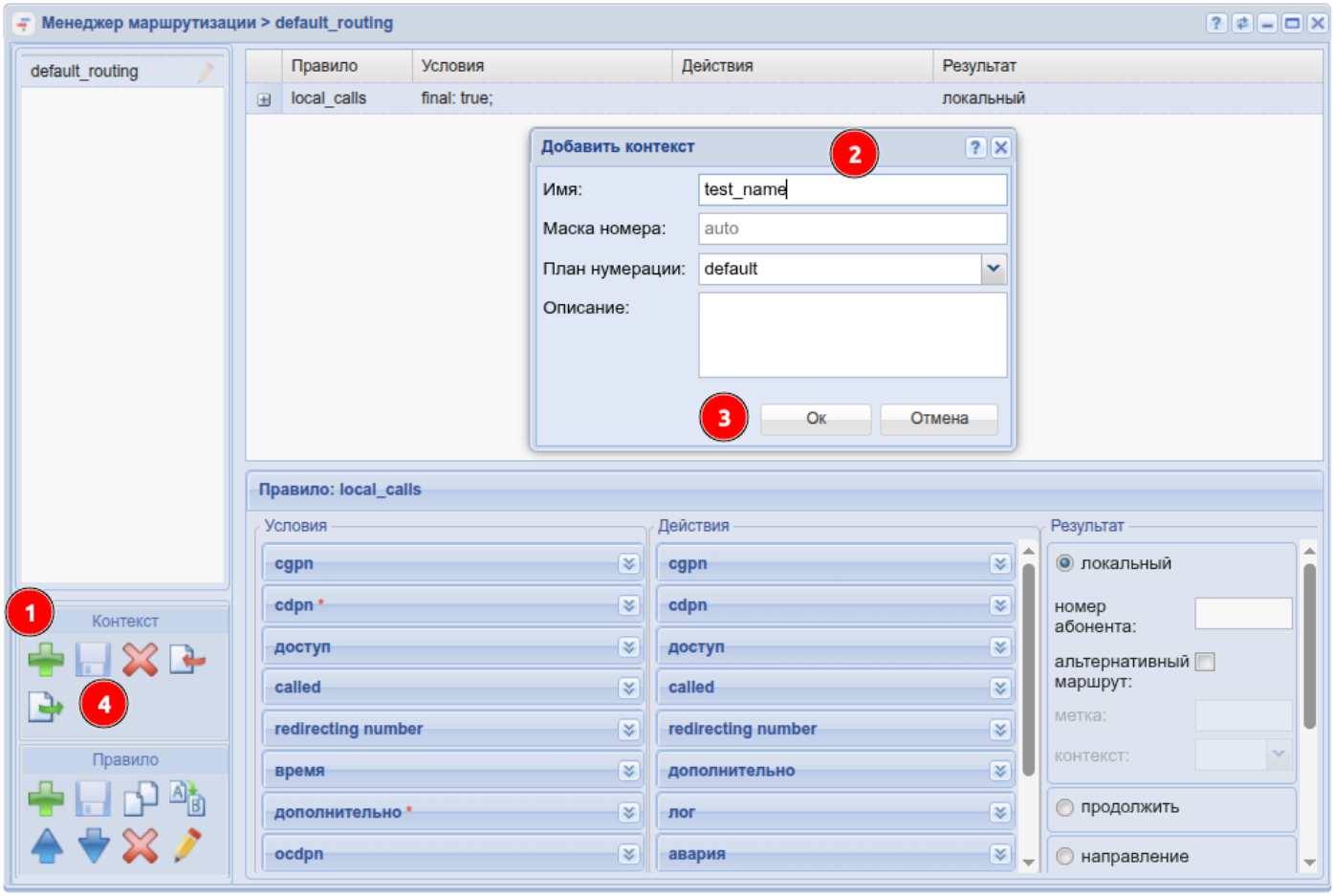


Рисунок 26 — Вид приложения "Менеджер маршрутизации" добавление нового контекста маршрутизации

3. Нажмите *"Сохранить контекст"*  ;

Создадим 4 правила в данном контексте:

- rule1 — правило для выхода на транк TAU-72;
- rule2 — правило локальной маршрутизации для номеров 42001-42005;
- rule3 — правило для выхода в ivr;
- rule4 — правило-исключение.

Для создания нового правила нужно выделить созданный контекст и нажать кнопку *"Создать правило"*. В появившемся окне нужно ввести название правила. Затем сохранить вновь созданные правила.

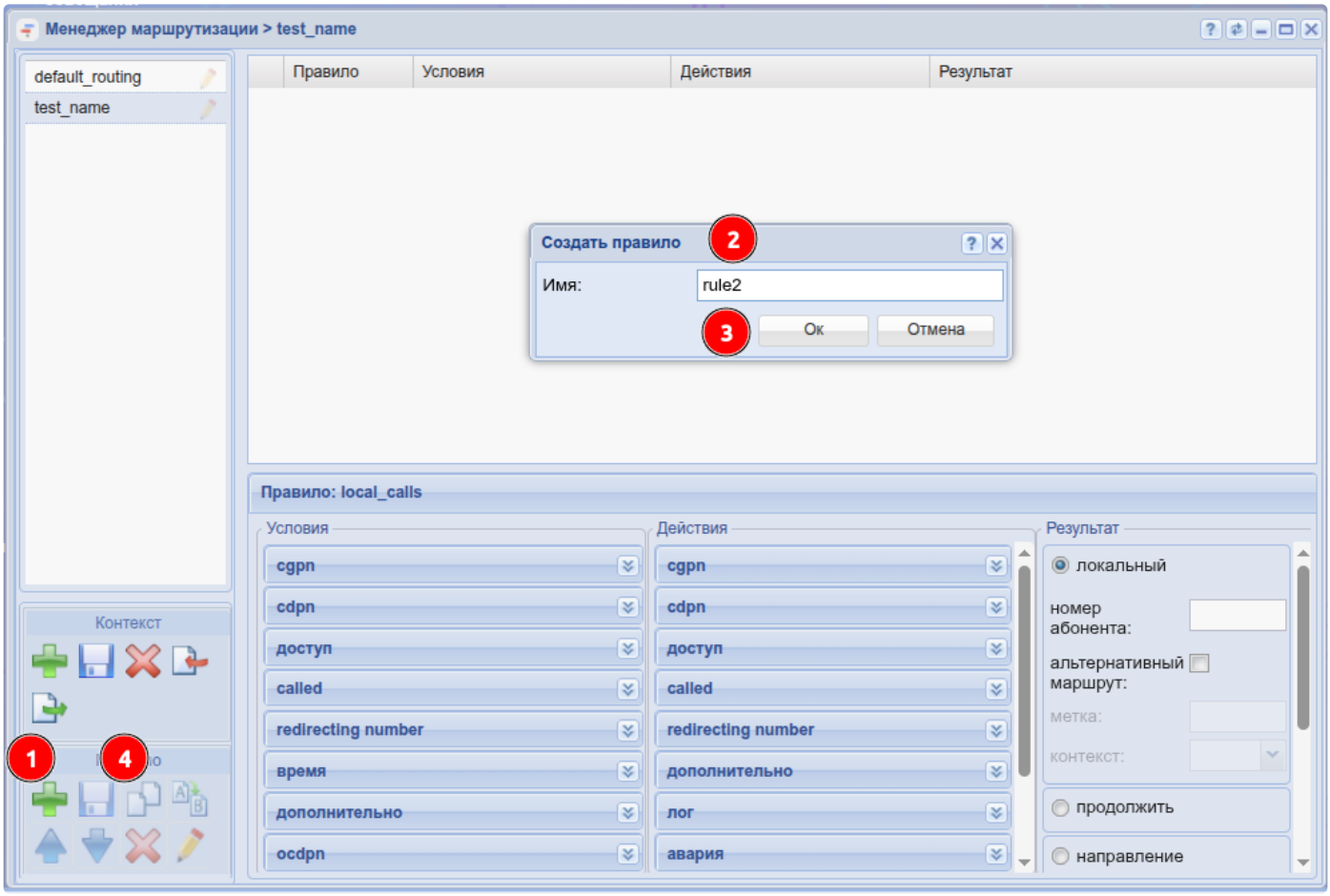


Рисунок 27 – Создание правила контекста маршрутизации

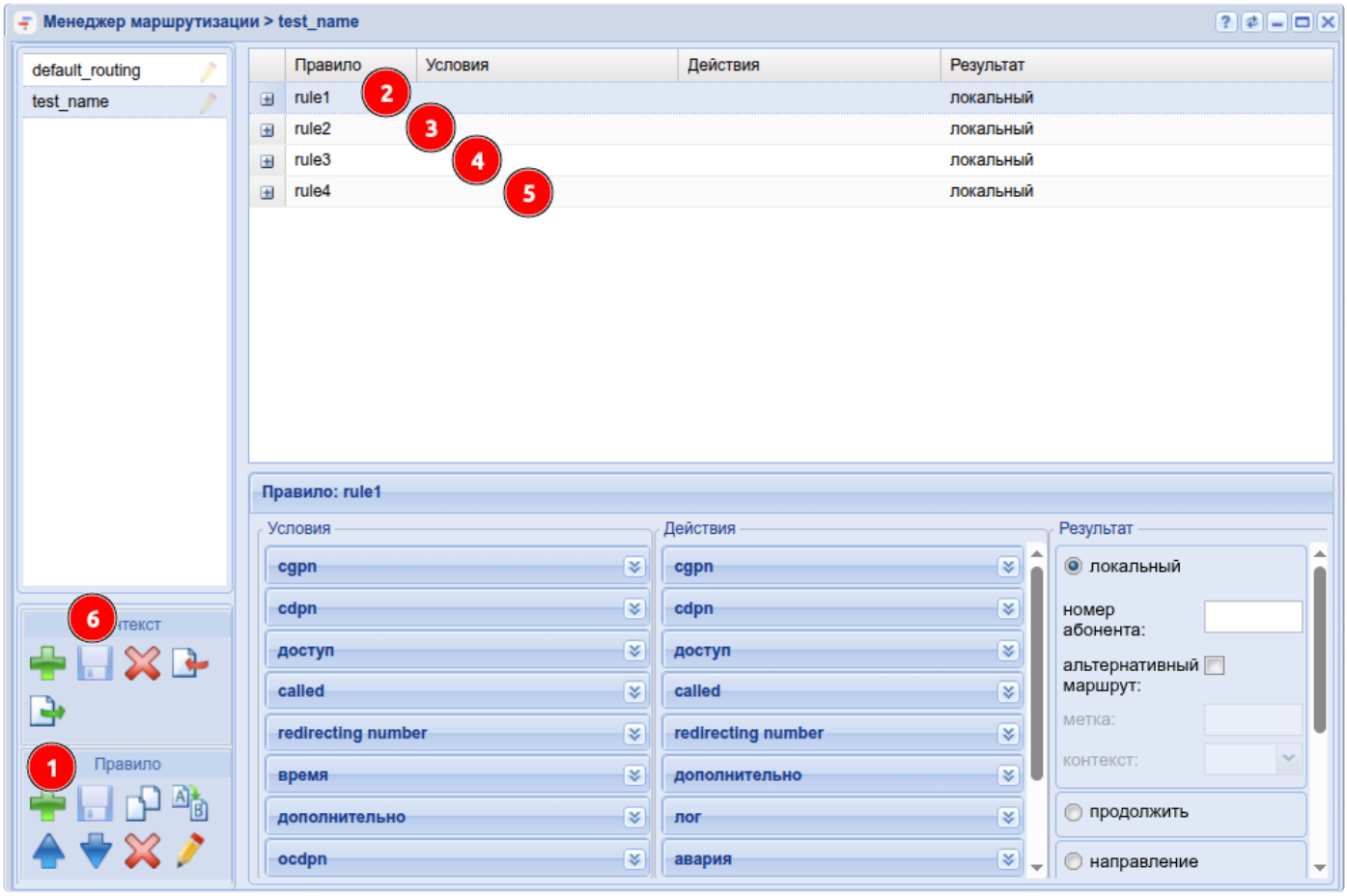


Рисунок 28 — Определение правил

На данный момент не определен транк, чтобы можно было сослаться на него в правиле, однако можно задать номера, по которым будет производится выборка.

Нажав на правило rule1, перейдем в нижнюю часть экрана, в которой располагаются области для редактирования контекста маршрутизации. Условно определим, что выборка для выхода на транк будет проводиться по признакам номера вызываемого абонента (CDPN), а номера в транке должны начинаться с цифры 4.

Функционально контекст маршрутизации делится на три части:

- Условия ("Condition") — раздел, определяющий выражения для выборки по предложенным признакам;
- Действия ("Action") — раздел, выполняющий преобразование признаков номеров к конкретному значению;
- Результат ("Result") — раздел, завершающий маршрутизацию и определяющий её результат.

rule1: Для выхода на транк нужно правильно отредактировать каждую часть:

- В разделе условий перейти на вкладку CDPN. Ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера с 42006 по 42007, нужно ввести условие: 420(6-7);
- В разделе действий перейти на вкладку CDPN. Ввести маску для изменения номера в поле "Номер". Например, чтобы добавить число 4 перед номером, нужно заполнить поле следующим выражением: 4{1,2,3,4,5};
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить транк в системе, поэтому к настройке данного правила вернемся чуть позже.

Таким же образом нужно настроить оставшиеся правила.

Для rule2:

- В разделе условий, во вкладке CDPN ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера со 42001 по 42005, ввести условие 420(1-5);
- В разделе результата определить результат как локальный (т.е. локальная маршрутизация);
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Для rule 3 допустим, что абоненты с номерами 108, 109 и 110 попадают в ivr-скрипт–информатор, прежде чем звонить далее.

- В разделе условий, во вкладке CGPN ввести номера телефонов, по которым будет выборка в поле "Номер", например 420(10,08-09).
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить в системе ivr-скрипт, поэтому к настройке данного правила вернемся чуть позже.

Для rule 4 мы определяем правило-исключение — это правило, которое работает в случае, если любые другие правила не срабатывают.

- По умолчанию данное правило создается в системе в контексте default_routing. В условиях на вызываемый номер прописывается символ %, а результатом является локальная маршрутизация, однако, если мы назначаем абоненту контекст, отличный от default_routing, в конце *рекомендуется* создавать это правило.

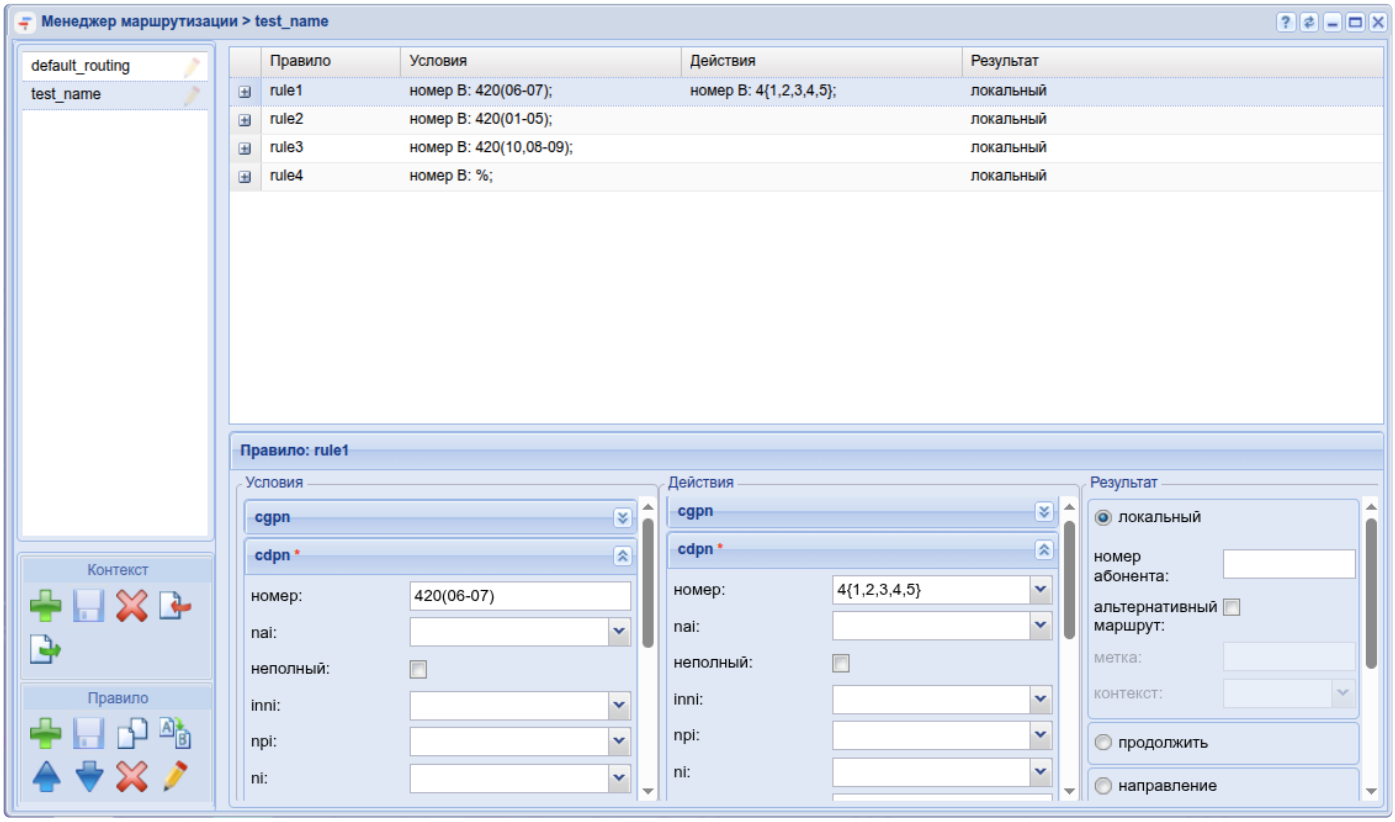


Рисунок 29 — Пример настроек контекста маршрутизации

8.10.9 Создание транка

Для создания и редактирования параметров транка в системе используется приложение "Менеджер транков" ("Trunk manager") .

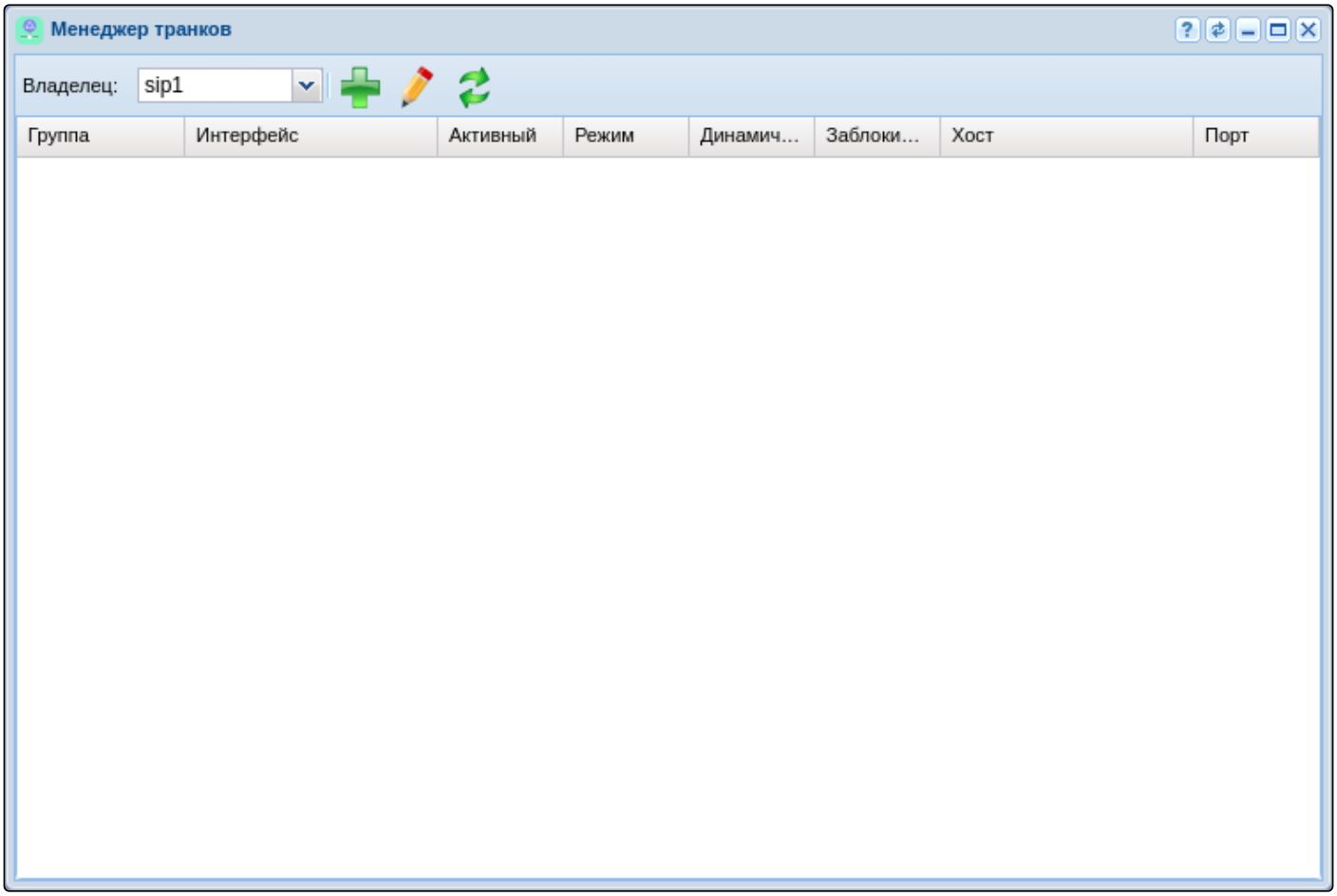


Рисунок 30 — Вид приложения "Менеджер транков"

Для определения транка в системе нужно нажать на кнопку *"Создать транк"* , в появившемся диалоговом окне определить параметры:

- *Имя* — назначить название транка, по которому его можно идентифицировать в системе;
- *Контекст* — применить ранее созданный контекст маршрутизации test_name;
- *Группа* — выбрать группу интерфейсов, созданную при определении абонентов "test.group";
- *Группа IP-адресов (IP-set)* — созданная на домене группа адресов "test_set";
- *Регистрация* — если используется транковая, то нужно включить данный параметр, в примере она не используется;
- *Хост [:порт]* — IP-адрес назначения транка — 10.0.3.100;
- *Порт для приёма* — транспортный порт, на котором будет слушаться трафик с транка, соответствует порту, назначенному на группу IP-адресов.

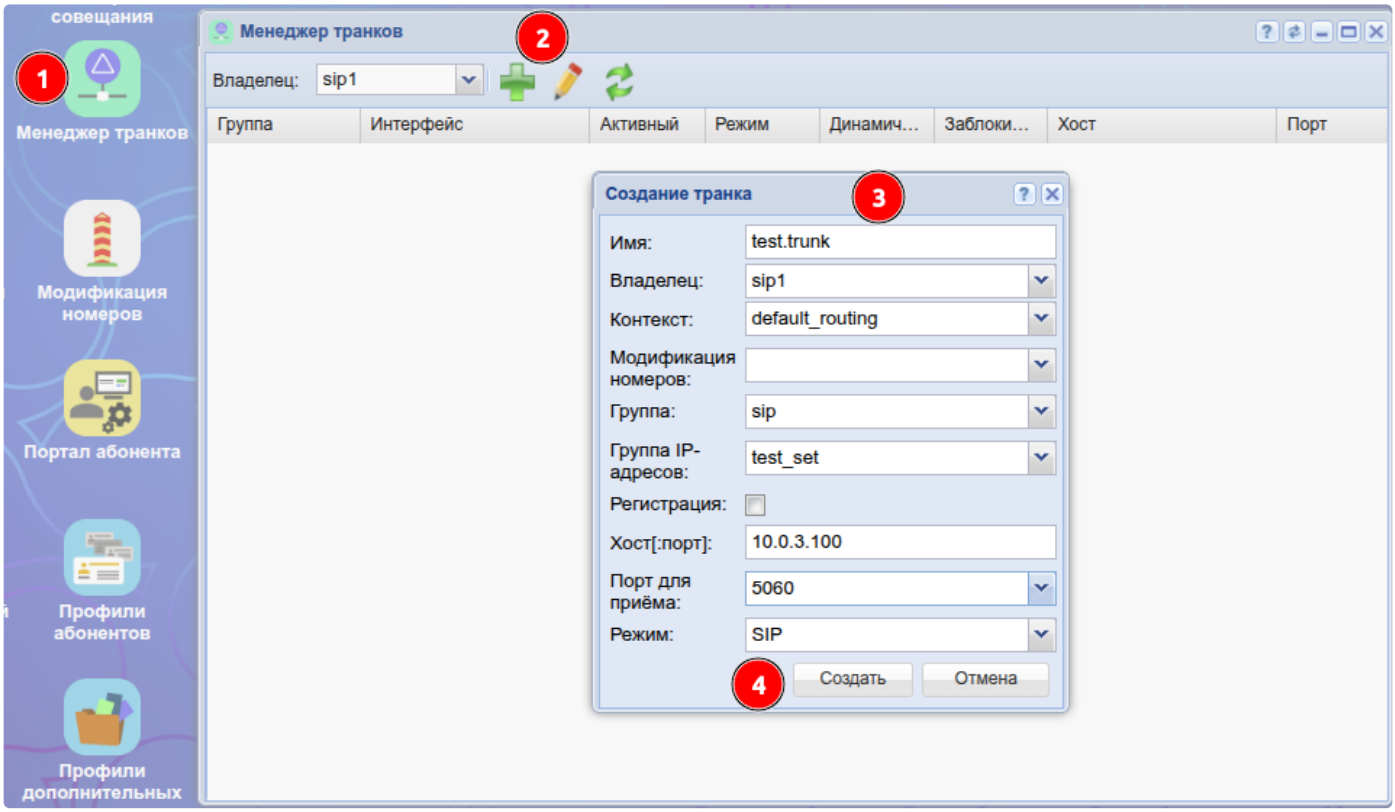


Рисунок 31 — Создание транка

8.10.10 Создание IVR-скрипта

Для создания IVR используется приложение "IVR-редактор" ("IVR editor").

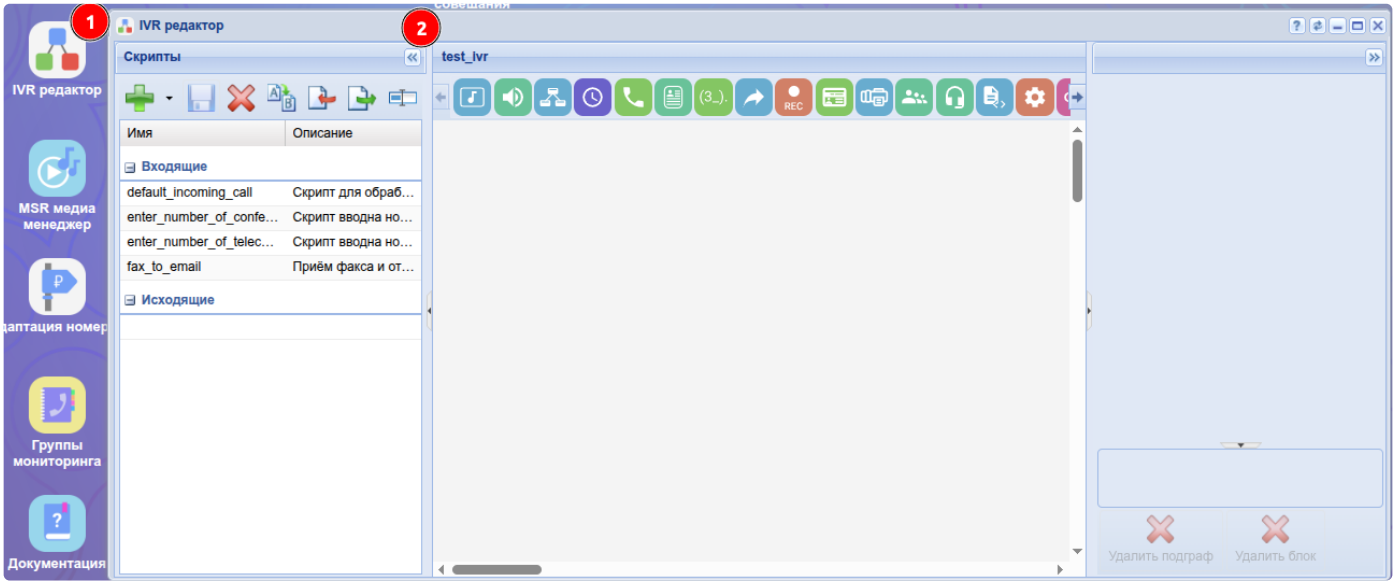


Рисунок 32 — Вид приложения "IVR-редактор"

Для создания скрипта нужно нажать на кнопку *"Добавить"* , выбрать тип скрипта (в данном случае будет скрипт на входящие вызовы), в диалоговом окне указать название скрипта, например "test_ivr".

После создания скрипта в основном окне редактора появится блок-схема.

В качестве примера составим скрипт, который при срабатывании проигрывает вызывающему абоненту заранее записанную фразу, а затем продолжает вызов.

Пример скрипта приведен на рисунках ниже. Здесь используются блоки Info и Dial.

- Блок Info — блок, который проигрывает сообщения до момента ответа пользователя. В качестве записанной фразы используется генератор тонов:

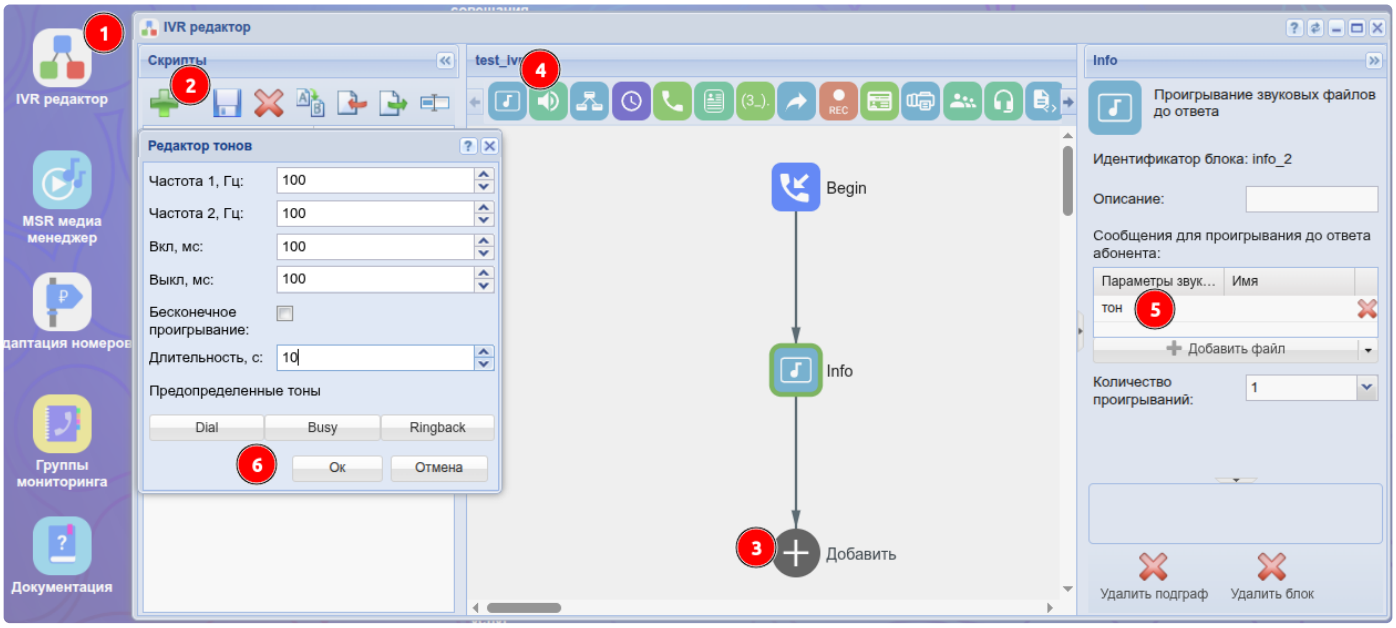


Рисунок 33 — Вид рабочей области IVR с настройками блока "Info"

- Блок dial — блок, который совершает вызов на заданный номер. Чтобы продолжить вызов, нужно применить предопределенную переменную CDPN:

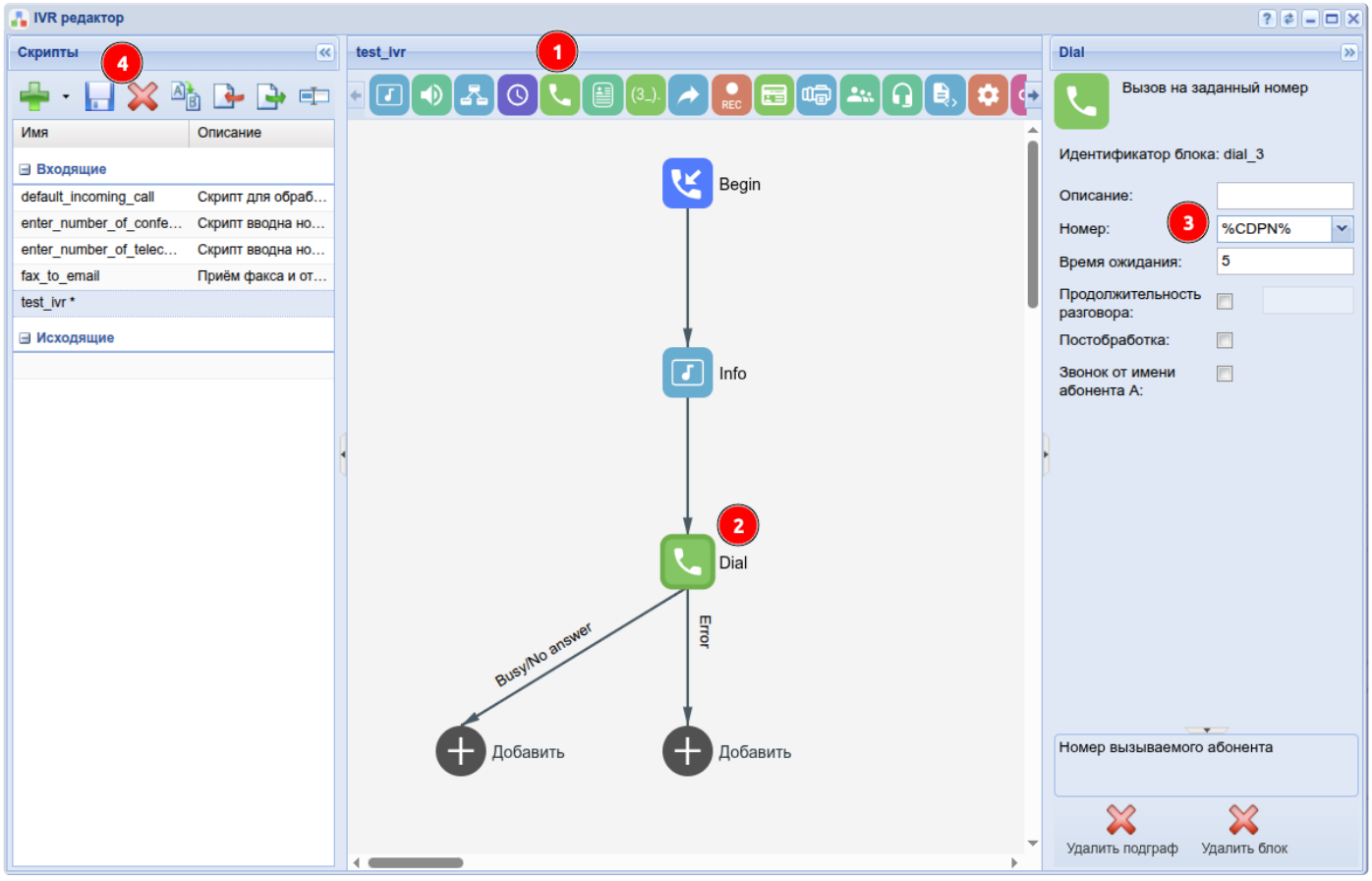


Рисунок 34 — Вид рабочей области IVR

8.10.11 Завершение настройки маршрутизации

Для завершения настройки маршрутизации нужно войти в приложение "Менеджер маршрутизации" ("Routing manager") и в правилах rule1 и rule3 настроить соответствующие результаты маршрутизации, для этого:

- rule1: в разделе "Результат" выберите подпункт "внешний". В таблице "Направления" в поле "Значение" добавьте созданный ранее транк, после чего сохраните правило .
- rule3: В разделе "Результат" выберите подпункт "ivr" и в поле скрипт добавьте созданный ранее ivr-скрипт, после чего сохраните правило и контекст .

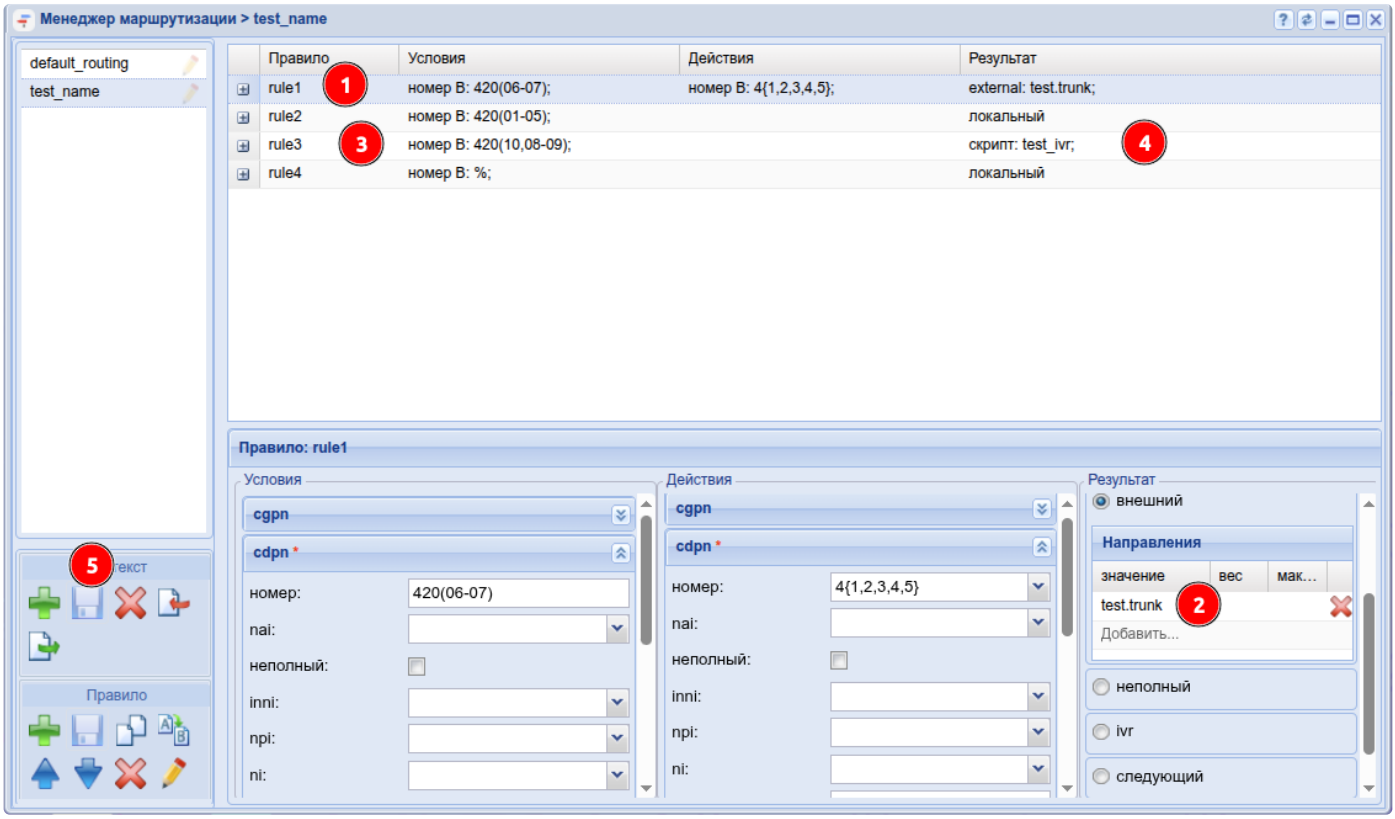


Рисунок 35 — Вид контекста маршрутизации

8.11 Первоначальная конфигурация системы через интерфейс командной строки (CLI)

пример первоначальной конфигурации через CoCon (CLI)

8.11.1 Добавление медиа-ресурсов в интерфейсе командной строки (CLI)

Медиаресурс — это описание параметров медиасервера, необходимых для работы с ним.

⚠ Для возможности использования медиаресурса и установления соединений , необходимо выполнить активизацию в команде создания медиаресурса - **/system/media/resource/declare** (параметр **"is active"=true**) или **/system/media/resource/set** (параметр **"active"=true**), если при создании активизация не была выполнена. Так же можно выполнить активизацию через веб интерфейс , приложение - сетевые окончания MSR (MSR-registrars).

declare

declare
Команда для объявления подключения медиасервера к ECSS-10.

Для работы в режиме предварительно настроенной системы можно заранее прописать медиаресурсы, информация о которых известна.

Для этого необходимо создать описание известных медиасерверов с указанием всех параметров, что позволить начать работу с медиасервером сразу после его регистрации.

Команда для создания конфигурации медиасервера:

Путь команды:

/system/media/resource/declare

Синтаксис:

declare <node> by_contact <contact> <zone> <site> <is active>

или

declare <node> by_iface <msr> <acc_id> <iface> <zone> <site> <is active>

Параметры:

<node> — имя ноды core;
<contact> — contact uri декларируемого медиаресурса;
<zone> — логическая зона обслуживания MSR;
<site> — идентификатор сети обслуживания MSR;
<is active> — состояние активности медиаресурса, значения true/false;
<msr> — имя MSR;
<acc_id> — номер идентификатор ресурса;
<iface> — название интерфейса.

set

set

Команда для изменения настроек медиаресурсов.

Путь команды:

/system/media/resource/set

Синтаксис:

set <node|*> <contact> <active|zone|site> <value>

Параметры:

<node|*> - имя ноды;
<contact> - contact uri декларируемого медиаресурса;
<active|zone|site> - активация | зона | сайт;
<value> - значение параметра.

Пример:

Активируем связь MSR- и Core-подсистем. Для этого пропишите команды:

выполняется в CoCon

```
/system/media/resource/set core1@ecss1 net.20@msr.ecss1 active true
```

выполняется в CoCon

```
/system/media/resource/set core1@ecss2 net.20@msr.ecss2  active true
```

выполняется в CoCon

```
/system/media/resource/set core1@ecss1 net.20@msr.ecss2  active true
```

выполняется в CoCon

```
/system/media/resource/set core1@ecss2 net.20@msr.ecss1  active true
```

```
/system/media/resource/set core1@ecss1 net.20@msr.ecss1 active true
  Media resource:
┌──────────────┬──────────────────┬──────────┬─────────┐
│     Node     │     Contact      │ Property │  Value  │
├──────────────┼──────────────────┼──────────┼─────────┤
│ core1@ecss1  │ net.20@msr.ecss1 │ active   │ true    │
└──────────────┴──────────────────┴──────────┴─────────┘
/system/media/resource/set core1@ecss2 net.20@msr.ecss2 active true
  Media resource:
┌──────────────┬──────────────────┬──────────┬─────────┐
│     Node     │     Contact      │ Property │  Value  │
├──────────────┼──────────────────┼──────────┼─────────┤
│ core1@ecss2  │ net.20@msr.ecss2 │ active   │ true    │
└──────────────┴──────────────────┴──────────┴─────────┘
/system/media/resource/set core1@ecss1 net.20@msr.ecss2 active true
  Media resource:
┌──────────────┬──────────────────┬──────────┬─────────┐
│     Node     │     Contact      │ Property │  Value  │
├──────────────┼──────────────────┼──────────┼─────────┤
│ core1@ecss1  │ net.20@msr.ecss2 │ active   │ true    │
└──────────────┴──────────────────┴──────────┴─────────┘
/system/media/resource/set core1@ecss2 net.20@msr.ecss1 active true
  Media resource:
┌──────────────┬──────────────────┬──────────┬─────────┐
│     Node     │     Contact      │ Property │  Value  │
├──────────────┼──────────────────┼──────────┼─────────┤
│ core1@ecss2  │ net.20@msr.ecss1 │ active   │ true    │
└──────────────┴──────────────────┴──────────┴─────────┘
```

info

info
Команда для просмотра информации о состоянии подключения конкретного медиаресурса.

Путь команды:

/system/media/resource/info

Синтаксис:

info <node> <contact>

Параметры:

<contact> — contact uri декларируемого медиаресурса;
<node> — имя ноды.

Пример:

```
/system/media/resource/info core1@ecss1 net.20@msr.ecss1
  Media resource specific for node core1@ecss1:
┌─┬──────────────────┬──────────────────────┐
│F│    Parameter     │        Value         │
├─┼──────────────────┼──────────────────────┤
│r│ Contact          │ net.20@msr.ecss1     │
│r│ Status           │ registered           │
│r│ Cc-status        │ connected            │
│r│ Cc-status uptime │ 00:39:35             │
│r│ Cc-id            │ db21af               │
│r│ Address          │ 10.0.20.51:5700      │
│r│ Register time    │ 2025/12/02 16:43:07  │
│r│ Expired          │ 84                   │
│r│ Iface            │ net.20(10.0.20.51)   │
│ │ Active           │ true                 │
│ │ Zone             │ default              │
│ │ Site             │ local                │
│r│ MSR version      │ 3.18.1.0.10          │
│r│ MSR load         │ 0                    │
│r│ MSR perf coef    │ 1.0                  │
└─┴──────────────────┴──────────────────────┘

Legend:
Column F - means Flag, values r - flag for readonly parameter
```

list

list
Команда для просмотра списка медиаресурсов в системе ECSS-10.

Путь команды:

/system/media/resource/list

Синтаксис:

list [active | all | register]

Параметры:

active — отображать только активные медиаресурсы (по умолчанию);
register — отображать только зарегистрированные медиаресурсы;
short — отобразить минимальную информацию;
all — отображать все задекларированные медиаресурсы.
По умолчанию команда выдает список зарегистрированных медиасерверов.

Пример:

```
/system/media/resource/list all
  All media resource selected list specific:
┌──────────────┬──────────┬──────────────┬──────────┬──────┬──────┬──────────┬──────────
──┬──────────┬──────────┐
│     Node     │    MSR   │     MSR      │    MSR   │ MSR  │Cc-id │Cc-status │Cc-uptime
│   Cc-address │ Iface    │    Iface     │ Active   │ Zone │Site  │    Contact    │
   Status  │ Expired  │
│              │          │   version    │perf coef │load  │      │          │
│                │  name  │    addr      │          │      │      │                │
│         │
├──────────────┼──────────┼──────────────┼──────────┼──────┼──────┼──────────┼──────────
──┼──────────┼──────────┤
│ core1@ecss1  │msr.ecss1 │3.18.1.0.10   │      1.0 │    0 │db21af│connected │00:40:22
│10.0.20.51:5700 │net.20  │10.0.20.51    │true      │default │local │net.20@msr.ecss1 │
 registered │     156  │
│              │msr.ecss2 │3.18.1.0.10   │      1.0 │    0 │263898│connected │00:40:22
│10.0.20.52:5700 │net.20  │10.0.20.52    │true      │default │local │net.20@msr.ecss2 │
 registered │     156  │
│ core1@ecss2  │msr.ecss1 │3.18.1.0.10   │      1.0 │    0 │db21af│connected │00:41:11
│10.0.20.51:5700 │net.20  │10.0.20.51    │true      │default │local │net.20@msr.ecss1 │
 registered │     106  │
│              │msr.ecss2 │3.18.1.0.10   │      1.0 │    0 │263898│connected │00:44:00
│10.0.20.52:5700 │net.20  │10.0.20.52    │true      │default │local │net.20@msr.ecss2 │
 registered │     177  │
└──────────────┴──────────┴──────────────┴──────────┴──────┴──────┴──────────┴──────────
──┴──────────┴──────────┘
```

8.11.2 Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI)

/system/media/registrar/ - команды управления регистратором для MSR медиа менеджера

set

set

Команда для изменения общих и индивидуальных параметров.

Путь команды:

/system/media/registrar/set

Синтаксис настройки общих параметров:

set common <FIELD> <VALUE>

Параметры:

<FIELD> - имя настраиваемого параметра;

<VALUE> - значение настраиваемого параметра

Где

- common - тип выставляемого параметра;

- <common parameter> - общий параметр медиарегистратора для всех нод: registration-timeout;
- private - тип выставляемого параметра;
- <node name> - имя ноды на которой находится медиарегистратор;
- <private parameter> - параметр медиарегистратора, который выставляется на <node name>: is-active, listen-interface & listen-port.

Список параметров доступных для настройки:

- is-active - позволяет включить или выключить регистратор (false - деактивирует регистратор, регистратор перестает принимать SIP трафик);
- listen-interface - имя интерфейса, который использует регистратор;
- listen-port - порт, на котором регистратор принимает запросы на регистрацию (по умолчанию 5000);
- registration-timeout - таймер перерегистрации, диапазон значений от 60 до 86400.

В общих параметрах доступна настройка времени регистрации, по умолчанию установлено 60 секунд.

Пример:
Установить время регистрации 60 секунд:

```
/system/media/registrar/set common registration-timeout 60
Common parameter registration-timeout set to 60.
```

Синтаксис настройки индивидуальных параметров:

system/media/registrar/set private <CORE> <FIELD> <VALUE>

Параметры:

<CORE> - нода ядра, для которой настраивается Registrar-сервер;

<FIELD> - имя настраиваемого параметра;

<VALUE> - значение настраиваемого параметра.

Настраиваемые параметры:

- is-active - активен ли Registrar-сервер, принимает значения true | false;
- listen-ip - ip-адрес, на котором Registrar-сервер будет принимать соединения (по умолчанию 0.0.0.0);
- listen-port - сетевой порт, на котором Registrar-сервер будет принимать соединения (по умолчанию 5000).

Установить ipадрес для приема соединений от медиасервера:

выполняется в CoCon

```
/system/media/registrar/set private core1@ecss1 listen-ip 10.0.20.51
```

выполняется в CoCon

```
/system/media/registrar/set private core1@ecss2 listen-ip 10.0.20.52
```

```
/system/media/registrar/set private core1@ecss1 listen-ip 10.0.20.51
Private parameter listen-ip set to 10.0.20.51 at node core1@ecss1.

/system/media/registrar/set private core1@ecss2 listen-ip 10.0.20.52
Private parameter listen-ip set to 10.0.20.52 at node core1@ecss2.
```

info

info

Команда просмотра текущих настроек Registrar-сервера.

Путь команды:

/system/media/registrar/info

```
/system/media/registrar/info
 Common parameters:
┌──────────────────────────┬─────────┐
│         Parameter        │ Value   │
├──────────────────────────┼─────────┤
│ registration-timeout, sec│   180   │
└──────────────────────────┴─────────┘
 Private parameters:
┌──────────────┬──────────┬────────────┬──────────────┬──────────────┐
│    Node      │ Status   │ Is-active  │ Listen-ip    │ Listen-port  │
├──────────────┼──────────┼────────────┼──────────────┼──────────────┤
│ core1@ecss2  │ active   │ true       │ 10.0.20.52   │        5000  │
│ core1@ecss1  │ active   │ true       │ 10.0.20.51   │        5000  │
└──────────────┴──────────┴────────────┴──────────────┴──────────────┘
```

После того как все сервисы будут запущены, ноды какое-то время будут устанавливать связь. Как только все ноды загрузятся , при выполнении запроса **system-status**, последует вывод следующей информации:

```
/system-status
Checking...
┌─┬──────────────┬──────────────────────────────┬──────────────────────────────────┬──────────────┬─────────┐
│ │     Node     │          Release             │         Erlang nodes             │ Mnesia nodes │ Uptime  │
├─┼──────────────┼──────────────────────────────┼──────────────────────────────────┼──────────────┼─────────┤
│ │core1@ecss1   │ecss-core-3.18.1.0.1656       │core1@ecss1,core1@ecss2           │not running   │43m 43s  │
│ │core1@ecss2   │ecss-core-3.18.1.0.1656       │core1@ecss1,core1@ecss2           │not running   │47m 5s   │
│ │ds1@ecss1     │ecss-ds-3.18.1.0.1656         │ds1@ecss1,ds1@ecss2               │ds1@ecss1,ds1@ecss2 │43m 43s  │
│ │ds1@ecss2     │ecss-ds-3.18.1.0.1656         │ds1@ecss1,ds1@ecss2               │ds1@ecss1,ds1@ecss2 │47m 4s   │
```

```
| |md1@ecss1      |ecss-mediator-3.18.1.0.1656|md1@ecss1,md1@ecss2
|md1@ecss1,md1@ecss2  |43m 43s|
| |md1@ecss2      |ecss-mediator-3.18.1.0.1656|md1@ecss1,md1@ecss2
|md1@ecss1,md1@ecss2  |47m 5s |
| |mycelium1@ecss1|ecss-mycelium-3.18.1.0.1656|mycelium1@ecss1,mycelium1@ecss2|not
running          |43m 43s|
| |mycelium1@ecss2|ecss-mycelium-3.18.1.0.1656|mycelium1@ecss1,mycelium1@ecss2|not
running          |47m 4s |
| |sip1@ecss1     |ecss-pa-sip-3.18.1.0.1656  |sip1@ecss1,sip1@ecss2
|sip1@ecss1,sip1@ecss2|43m 43s|
| |sip1@ecss2     |ecss-pa-sip-3.18.1.0.1656  |sip1@ecss1,sip1@ecss2
|sip1@ecss1,sip1@ecss2|47m 5s |
└─┴───────────────┴──────────────────────────┴──────────────────────────────────┴────
──────────────┴────────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────────┬────────────┬─────────────┐
│    Node      │    MSR   │     MSR      │ Cc-status  │ Cc-uptime   │
│              │          │   version    │            │             │
├──────────────┼──────────┼──────────────┼────────────┼─────────────┤
│ core1@ecss1  │ msr.ecss1 │ 3.18.1.0.10 │ connected  │ 00:42:49    │
│              │ msr.ecss2 │ 3.18.1.0.10 │ connected  │ 00:42:48    │
│ core1@ecss2  │ msr.ecss1 │ 3.18.1.0.10 │ connected  │ 00:43:38    │
│              │ msr.ecss2 │ 3.18.1.0.10 │ connected  │ 00:46:27    │
└──────────────┴──────────┴──────────────┴────────────┴─────────────┘

Alarms list on md1:
┌──────────────┬──────────────┬──────────────┬──────────────────────┬──────────────
──────┬──────────────────────┬──────
─────┐
│     Date     │   Severity   │   Location   │       Cause          │       Class
│      Instance      │                          Message                          │
├──────────────┼──────────────┼──────────────┼──────────────────────┼──────────────
──────┼──────────────────────┼──────
─────┤
└──────────────┴──────────────┴──────────────┴──────────────────────┴──────────────
──────┴──────────────────────┴──────
─────┘
Selected: 0
Total: 0
```

В примере выше видно, что ноды вошли в кластер и *MSR* зарегистрировались на ноде *ecss-core*.

8.11.3 Создание и настройка группы SIP-IP-адресов (IP-set)

Произведите настройку адаптера SIP согласно ТЗ:

выполняется в CoCon

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss1 ip = 10.0.20
.31
```

выполняется в CoCon

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss2 ip = 10.0.20
.32
```

выполняется в CoCon

```
/cluster/adapter/sip1/sip/network/set ip_set test_set listen-ports list = [5060]
```

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss1 ip
= 10.0.20.31
Property "ip_set" successfully changed from:

   to
test_set: no ports set
test_set: sip1@ecss1 10.0.20.31
test_set: dscp 0.


/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss2 ip
= 10.0.20.32
Property "ip_set" successfully changed from:
test_set: no ports set
test_set: sip1@ecss1 10.0.20.31
test_set: dscp 0
   to
test_set: no ports set
test_set: sip1@ecss1 10.0.20.31
test_set: sip1@ecss2 10.0.20.32
test_set: dscp 0.


/cluster/adapter/sip1/sip/network/set ip_set test_set listen-ports list = [5060]
Property "ip_set" successfully changed from:
test_set: no ports set
test_set: sip1@ecss1 10.0.20.31
test_set: sip1@ecss2 10.0.20.32
test_set: dscp 0
   to
test_set: 5060
test_set: sip1@ecss1 10.0.20.31
test_set: sip1@ecss2 10.0.20.32
```

```
test_set: dscp 0.
```

8.11.4 Создание Домена и подключение IP-set

Создайте Домен (виртуальную АТС) и присвойте ему созданную выше группу (IP-set) настроек SIP-адаптера:

выполняется в CoCon

```
/domain/declare test_domain --add-domain-admin-privileges --add-domain-user-privileges
```

выполняется в CoCon

```
/domain/test_domain/sip/network/set ip_set [test_set]
```

```
/domain/declare test_domain --add-domain-admin-privileges --add-domain-user-privileges
New domain test_domain is declared


/domain/test_domain/sip/network/set ip_set [test_set]
Property "ip_set" successfully changed from:
[]
   to
["test_set"].
```

После создания домена, администратор системы должен открыть доступ на управление услугами оператору Домена (виртуальной АТС). Для этого использовать команду разрешения оператору Домена (виртуальной АТС) "test_domain" управлять всеми услугами :

выполняется в CoCon

```
/cluster/storage/ds1/ss/access-list add test_domain *
```

```
/cluster/storage/ds1/ss/access-list add test_domain *
Supplementary services successfully added in the domain test_domain.
```

Для возможности применения пакета услуг у абонентов сразу после создания, необходимо настроить политику применения лицензионных пакетов услуг (детальную информацию вы можете посмотреть здесь)

Создать политику применения лицензии с именем -"new_sub"

выполняется в CoCon

```
/domain/test_domain/ss/licence/politics/declare new_sub "Задаются при создании абонента"
```

```
/domain/test_domain/ss/licence/politics/declare new_sub "Задаются при создании
абонента"
Licence politic new_sub successfully declared.
```

Задать имя пакета услуг дополнительных сервисов, для этого использовать следующие команды :

выполняется в CoCon

```
/cluster/storage/ds1/licence/show-licence 1
```

нас интересует следующая информация :

```
/cluster/storage/ds1/licence/show-licence 1
SSW ID                                                  < Имя SSW ID >
Description                                             Лицензия для тестирования
Локация 2.

. . .

Supplementary Services licence package(s)
 name                                                   'ECSS-ADV'
 limit                                                  50000
 description                                            "Весь пакет услуг"
 SS list                                                [1,2,3,4,5]

. . .
```

имя пакета услуг в примере "**ECSS-ADV**".
Добавить в созданную политику пакет лицензионных услуг с именем "ECSS-ADV"(именно так он описан в лицензии):

выполняется в CoCon

```
/domain/test_domain/ss/licence/politics/package-add new_sub ECSS-ADV
```

```
/domain/test_domain/ss/licence/politics/package-add new_sub ECSS-ADV
Licence packages ["ECSS-ADV"] successfully added to politic new_sub.
```

Пакет добавлен, но в настоящий момент политика не активна (не может быть применена к абоненту). Для активации политики new_sub выполнить следующую команду:

выполняется в CoCon

```
/domain/test_domain/ss/licence/politics/activate new_sub
```

```
/domain/test_domain/ss/licence/politics/activate new_sub
Licence politic new_sub successfully activated.
```

Теперь при создании абонентов можем использовать дополнительные услуги.

9 Первоначальная установка системы 3.18 без резервирования с одним сервером

- Исходные данные
- Включение в сеть
- Последовательность установки SSW ECSS10 на одном сервере
  - Подготовительный этап
    - Предварительные требования
- Установка ОС
- Настройка сети
  - Подготовка сетевых интерфейсов системы
- "Оптимизация" Операционной Системы
  - Выставить параметры ОС в режим производительности
  - Отключить SWAP
  - Установка часового пояса
  - Улучшение работы высоконагруженных серверов
- Обновление программного обеспечения операционной системы
- Установка и настройка программного обеспечения
  - Настройка пакета ecss-dns-env
  - Установка Postgres
  - Установка пакета ecss-postgres-bdr-ssw
  - Установка ecss-node
    - Установка на сервере ecss1
  - Установка лицензии
    - Конфигурация LM
    - Подключение SSW к Менеджера Лицензий
  - Установка пакетов ecss-restf, ecss-media-server, ecss-web-conf
    - ecss-restfs
    - ecss-media-server
    - ecss-web-conf
  - Настройка snpmd
  - Настройка безопасности. SSH
  - Добавление услуг
- Первоначальная конфигурация системы с помощью web-конфигуратора
  - Исходные данные
  - Подготовка к работе
  - Создание учетной записи оператора
  - Добавление медиа-ресурсов.
  - Создание IP-set (sip-транспорта) и присвоение его к домену
  - Создание домена
  - Создание абонентов
  - Создание и применение контекстов маршрутизации для домена
  - Создание транка
  - Создание IVR-скрипта
  - Завершение настройки маршрутизации
- Первоначальная конфигурация системы через интерфейс командной строки (CLI)
  - Добавление медиа-ресурсов в интерфейсе командной строки (CLI)
  - Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI) CoCon
  - Создание и настройка группы SIP-IP-адресов (IP-set)
  - Создание Домена и подключение IP-set

9.1 Исходные данные

Исходные данные
В данном разделе использованы специфичные для данного продукта термины, ознакомиться с ними можно здесь.
Требуется интеграция программного коммутатора 5 класса ECSS-10 (SSW) на 1 физическом сервере с поддержкой SIP со следующими параметрами на нагрузку:

- Максимальное количество абонентов – 15000 (MUL — Max user limit);
- Максимальное количество одновременных соединений — 2000 (MCL — Max call limit);
- Резервирование системы не требуется;
- Количество сетевых интерфейсов ethernet — 4.

По техническому заданию требуется определить аппаратную платформу.

Актуальная информация с требованиями к серверам, которые должны использоваться для работы комплекса ECSS-10.

№	Требования к серверам SSW	Серия Light+	Серия Midi	Серия Heavy	Серия Супер Heavy	Серия Top (1)
1	**Характеристики системы**					
1.1	Максимальное количество абонентов	5000	10000	20000	40000	-
1.2	Максимальная нагрузка одновременных соединений class 5	800	1500	3000	6000	-
1.3	Максимальная нагрузка одновременных соединений class 4	2400	4500	9000	20000	80000
2	**Характеристики сервера**					
2.1	Марка	HP (Lenovo)	HP (Lenovo)	HP (Lenovo)	HP (Lenovo)	HP /Lenovo / Depo
2.2	Серия	DL20 Gen10/DL 360 Gen10 (SR530)	DL360 Gen10 (SR530/SR630)	DL360 Gen10 (SR630)	DL360 Gen10 (SR630)	DL360 Gen10 / SR650 V2 / Storm 3450
2.3	Процессор	Intel Xeon 4214	Intel Xeon 5220	Intel Xeon 6240	Intel Xeon 8268	Intel Xeon 8380
2.4	Количество процессоров	1	1	2	2	2

№	Требования к серверам SSW	Серия Light+	Серия Midi	Серия Heavy	Серия Супер Heavy	Серия Top (1)
2.5	Оперативная память	16 Gb	24 Gb	32 Gb	64 Gb	1024 Gb
2.6	HDD	От 3X500 SATA (от 7200 rpm)	От 3x300 Gb SAS (от 10000 rpm)	От 3x600 Gb SAS (от 10000 rpm) 2x150Gb SSD	От 6x800 Gb SSD, 2x300Gb M.2 SSD	От 6x1.2 Tb SSD, 2x300Gb M.2 SSD
2.7	RAID	no raid board	HW Raid, от 1Gb cache+battery	HW Raid, от 1Gb cache+battery	HW Raid, от 2Gb Flash cache, поддержка RAID-5	HW Raid, от 2Gb Flash cache, поддержка RAID-5
3	**Дополнительные комплектующие сервера (в базовый комплект не входят)**					
3.1	Лицензия на удалённое управление	по желанию	+	+	+	+
3.2	Резервный источник питания	по желанию	+	+	+	+
3.3	Организация хранилища записей разговоров	Дополнительные HDD, объединенные в RAID-5	Лицензия на HW Raid с поддержкой RAID-5, дополнительные HDD для хранения записей	Лицензия на HW Raid с поддержкой RAID-5, дополнительные HDD для хранения записей	Лицензия на HW Raid с поддержкой RAID-5, дополнительные HDD для хранения записей	Лицензия на HW Raid с поддержкой RAID-5, дополнительные HDD для хранения записей

Полоса пропускания СПД

Требуемая полоса пропускания СПД **не менее** 1000Mb/s

Минимальные требования при установке ECSS-10 на виртуальные машины*

№	Требования к серверам SSW	Уровень Light (2)(3)	Уровень Light+	Уровень Midi	Уровень Heavy	Уровень Супер Heavy	Уровень Top (1)
1	**Характеристики системы**						

№	Требования к серверам SSW	Уровень Light (2)(3)	Уровень Light+	Уровень Midi	Уровень Heavy	Уровень Супер Heavy	Уровень Top (1)
1.1	Максимальное количество абонентов	1000	5000	10000	20000	40000	-
1.2	Максимальная нагрузка одновременных соединений class 5	250	800	1500	3000	6000	-
1.3	Максимальная нагрузка одновременных соединений class 4	1000	4500	9000	20000	80000	-
2	**Характеристики сервера**						
2.4	Количество ядер/потоков (vCore)	**8C**	**12C/24T**	**18C/36T**	**36C/72T**	**48C/96T**	**80C/160T** и более
2.5	Частота процессора	от 2.4 Ghz	от 3 Ghz	от 3Ghz	от 3.6 Ghz	от 3.4 Ghz	от 3.2 Ghz
2.6	Оперативная память	8 Gb	16 Gb	24 Gb	32 Gb	от 64 Gb	от 512 Gb
2.7	HDD	От 50 GB, 75 IOPS	От 500 GB SATA (от 7200 rpm, от 75 IOPS)	От 500 Gb SAS (от 10000 rpm, 170 IOPS)	От 500 Gb SAS (от 10000 rpm, 170 IOPS) и 150Gb SSD	от 800 Gb SSD и 300Gb M.2 SSD	От 1.2 Tb SSD и 300Gb M.2 SSD

C - core, они же ядра

T - thread = поток, vCore

(1) — серия Top на текущий момент рассчитана для реализации контроллера транковых шлюзов (работа по 4 классу).
(2) — уровень Light применяется только для расчета виртуальных машин, для физических серверов минимальная рекомендуемая конфигурация — Light+.
(3) — уровень Light является минимальным рекомендуемым. При запросе расчета ВМ для малого числа абонентов (до 1000: 100, 200...) используем его.

*Если на системе планируется использование таких сервисов как КЦ, запись разговоров, видео-вызовы, то необходимо согласовать требования с технической поддержкой

Пример составления аппаратных требований:

Устройство	Требуемый ресурс		Серия аппаратного продукта
	MCL	MUL	
Сервер 1	2500	15000	Heavy

После определения по требованиям проекта составляем предварительную сетевую карту.

Пример разнесения составляющих по адресному пространству для одной ноды:

Имя сервера (хоста)	Роль	Интерфейс	Адрес	Адрес вашего проекта	Порт
Статические адреса программного коммутатора					
ecss1	Адрес сервера	net.10 (mgm)	10.0.10.10/24		-
ecss1	Адрес протокольного адаптера	net.20 (voip)	10.0.20.10/24		-
ecss1	Адрес шлюза	net.10 (mgm)	10.0.10.1		-
ecss1	Адреса DNS-серверов	net.10 (mgm)	10.0.10.1, 8.8.8.8		-
Внутренние адреса программного коммутатора					
ecss1	Адрес ядра (*ecss-core*)	lo	127.0.0.1/24		5000
ecss1	Адрес медиасервера (*ecss-media-server (MSR)*)	lo	127.0.0.1/24		5040

9.2 Включение в сеть

Топологию включения сервера в сеть для обеспечения резервирования рекомендуется делать с использованием 2-х коммутаторов.

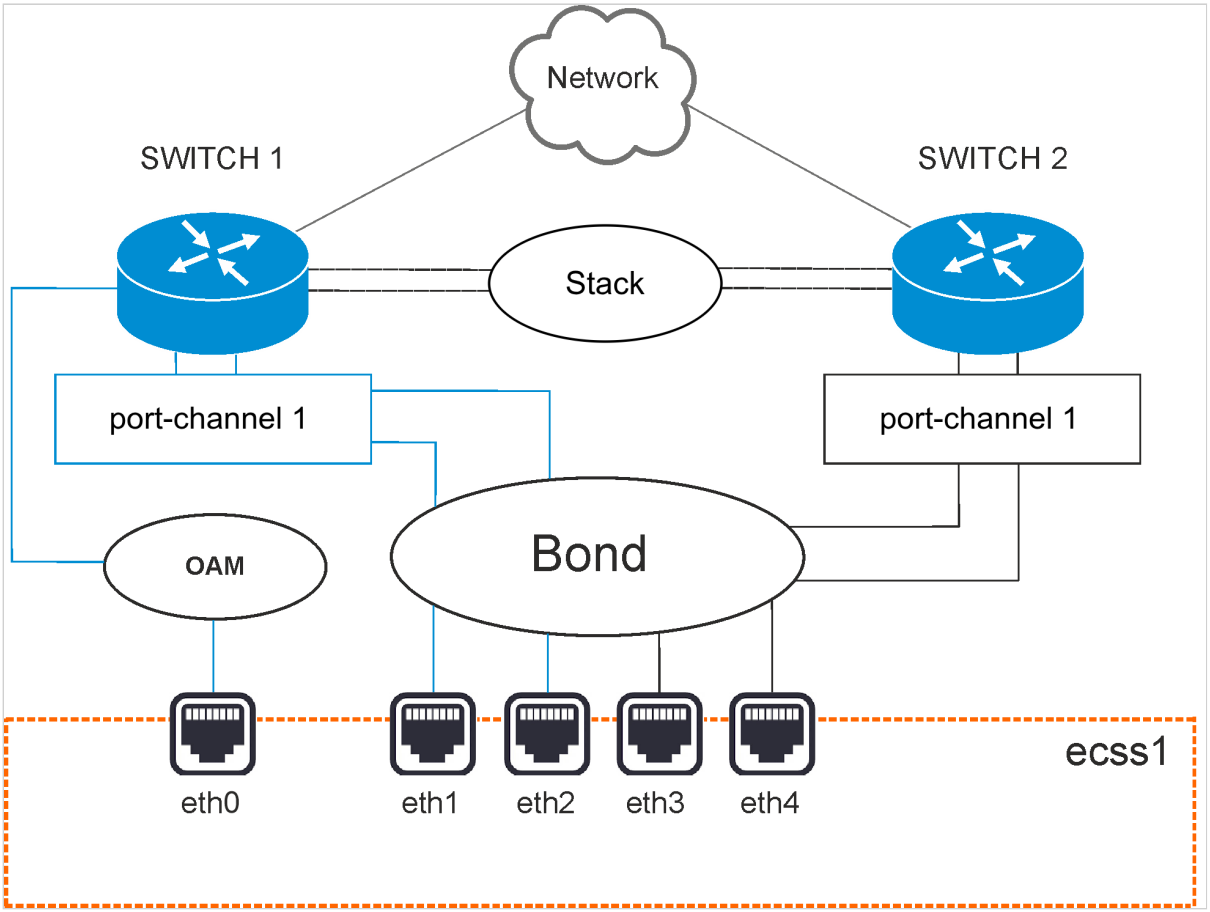


Рисунок 1 — Схема организации связи

Вариант 1. Схема active-backup

Коммутаторы связаны между собой в erps-кольцо.

Все 4 физических сетевых интерфейса объединяются в 1 агрегированный линк (bond). Агрегация портов сервера настраивается в режиме active-backup, т.е. в работе всегда находится только 1 сетевой интерфейс. Сетевые интерфейсы сервера попарно включены в коммутаторы, на которых агрегация портов (port-channel) также настраивается в режиме active-backup.

Например, eth1 и eth2 включаются в первый коммутатор, а eth3 и eth4 — во второй.

Вариант 2. Схема LACP

Коммутаторы связаны между собой в стек. Стек должен логически работать как один коммутатор, способный обеспечивать агрегацию портов в режиме LACP между разными физическими коммутаторами. Пример — коммутаторы MES3124 со специализированной прошивкой.

Все 4 физических сетевых интерфейса также объединяются в 1 агрегированный линк (bond). Агрегация портов сервера настраивается в режиме **802.3ad**. Создаются агрегированные группы сетевых карт с одинаковой скоростью и дуплексом. При таком объединении передача задействует все каналы в активной агрегации согласно стандарту IEEE 802.3ad. Выбор, через какой интерфейс отправлять пакет, определяется политикой. По умолчанию это XOR-политика, можно использовать «xmit_hash» политику. Подробнее в разделе Приложение E. Netplan.

Требования:

- Поддержка Ethtool в драйвере для получения информации о скорости и дуплексе на каждом сетевом интерфейсе;
- Поддержка на коммутаторе стандарта IEEE 802.3ad;

Сетевые интерфейсы сервера также попарно включены в коммутаторы, на которых агрегация портов (port-channel) настраивается в режиме LACP. Например, eth1 и eth2 включаются в первый коммутатор (port-channel 1), а eth3 и eth4 — во второй (port-channel 2).

9.3 Последовательность установки SSW ECSS10 на одном сервере

Установка SSW ECSS10 состоит из 2-х основных частей :

- Подготовительной - Установка ОС Ubuntu 22.04, обновлений пакетов ОС Ubuntu , "оптимизация" ОС, установка дополнительных пакетов программ, подготовка сетевых интерфейсов, /etc/hosts, ssh-keygen
- Установка приложений Eltex ECSS10

9.3.1 Подготовительный этап

В этом разделе приведено описание инсталляции операционной системы, а также необходимых и дополнительных пакетов. Система ECSS-10 версии 3.18 работает под управлением ОС Ubuntu 22.04.

Предварительные требования

- Установочный загрузочный носитель с дистрибутивом операционной системы;
- Подготовленный сервер с обновленным BIOS, ILO (если есть), подключенная сеть для доступа в сеть Интернет;
- Выставленный первый приоритет загрузки с установочного носителя — USB Flash или CD/DVD в BIOS;
- Достаточный объем дискового пространства и памяти в соответствии с проектом.

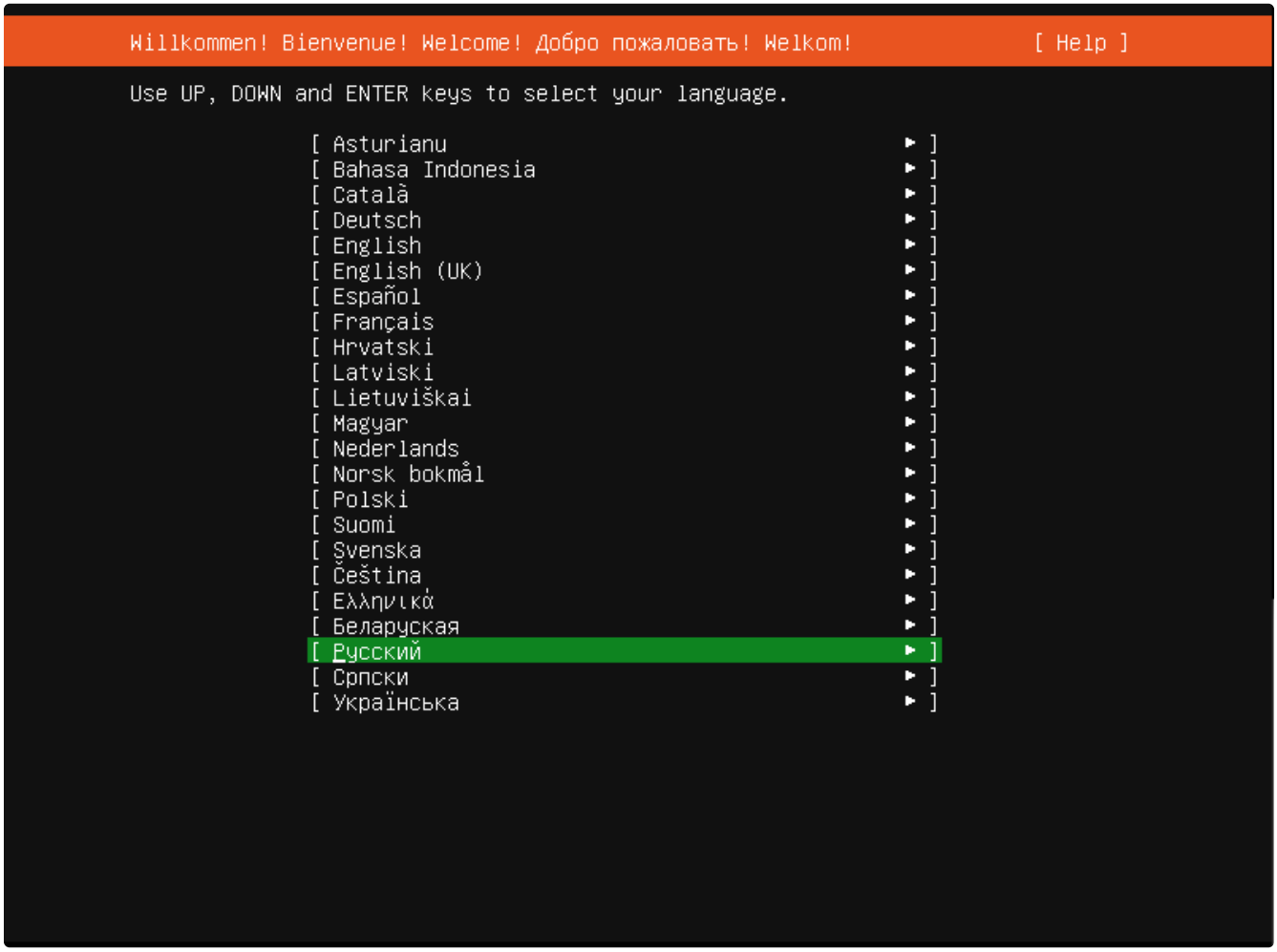
9.4 Установка ОС

Для установки ОС необходимо выполнить следующее:

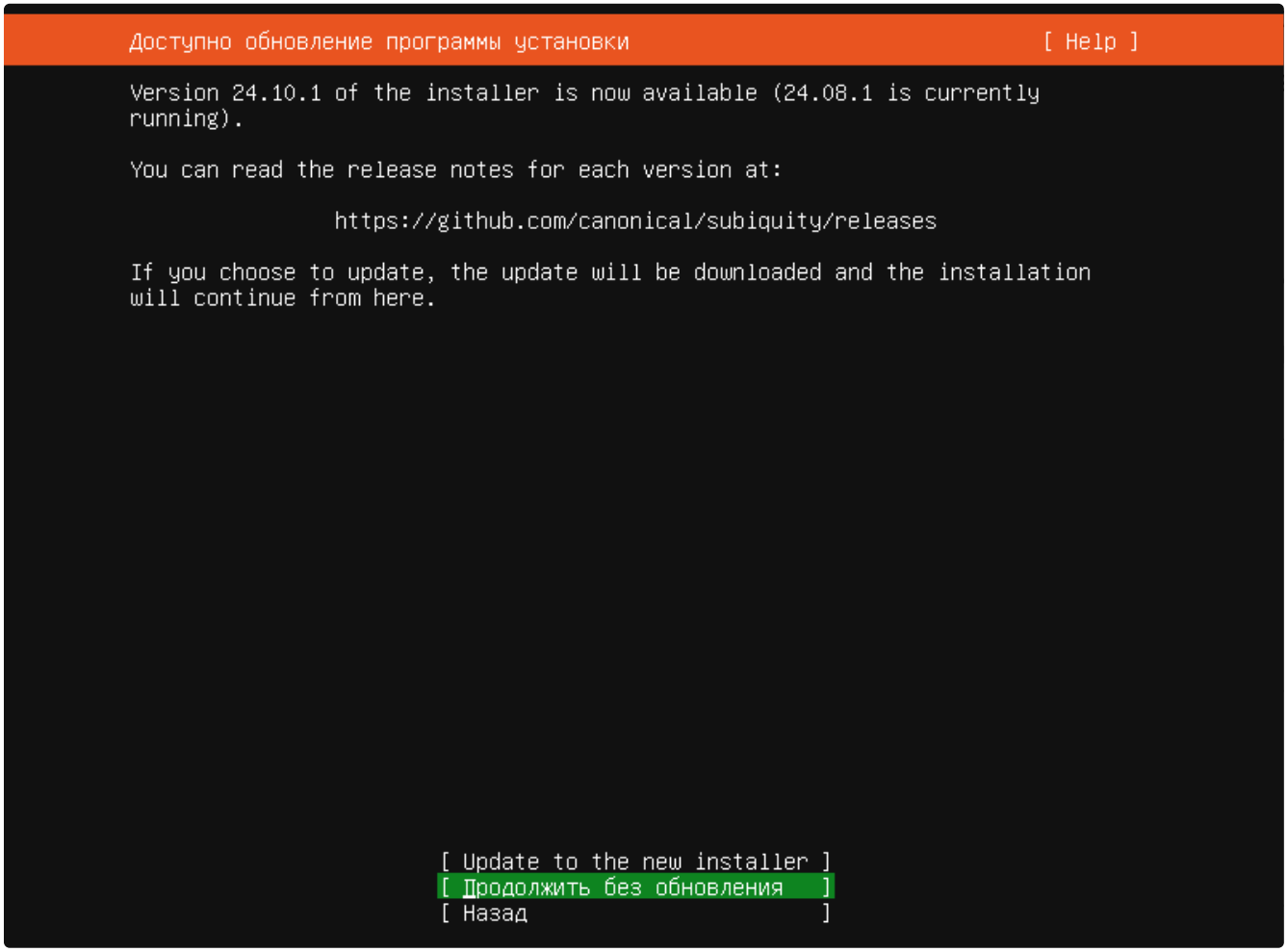
- После загрузки с установочного носителя выбрать "**Install Ubuntu Server**".
- Выбрать язык системы и раскладку клавиатуры.
- Настроить сетевой интерфейс для подключения к сети Интернет:
- Создать партиции на диске в соответствии с таблицей 1
- Настроить имена сервера и пользователя
- Установить OpenSSH server
- Перезагрузить успешно установленную ОС

Установка ОС Ubuntu Server 22.04

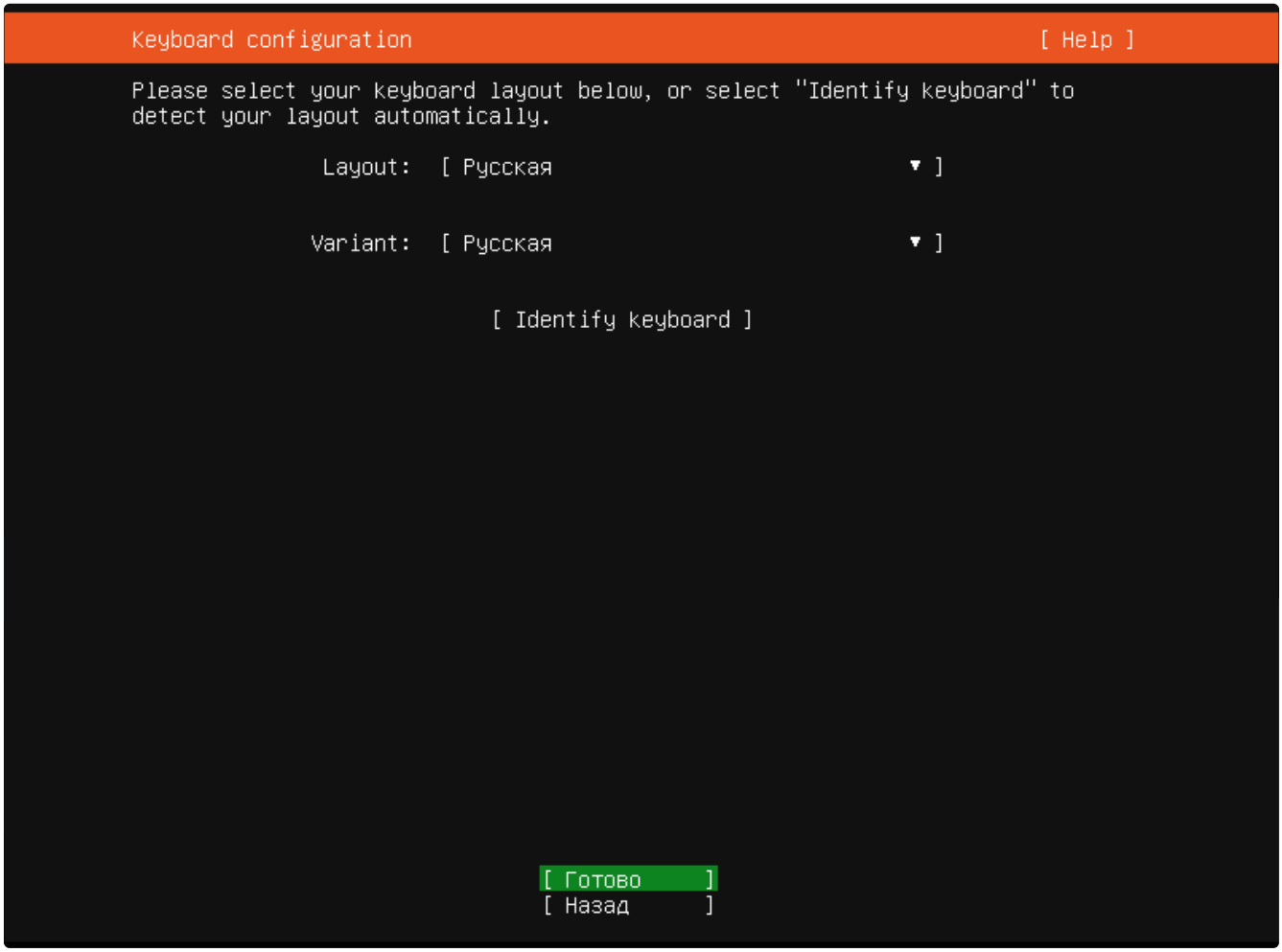
Выбираем язык интерфейса



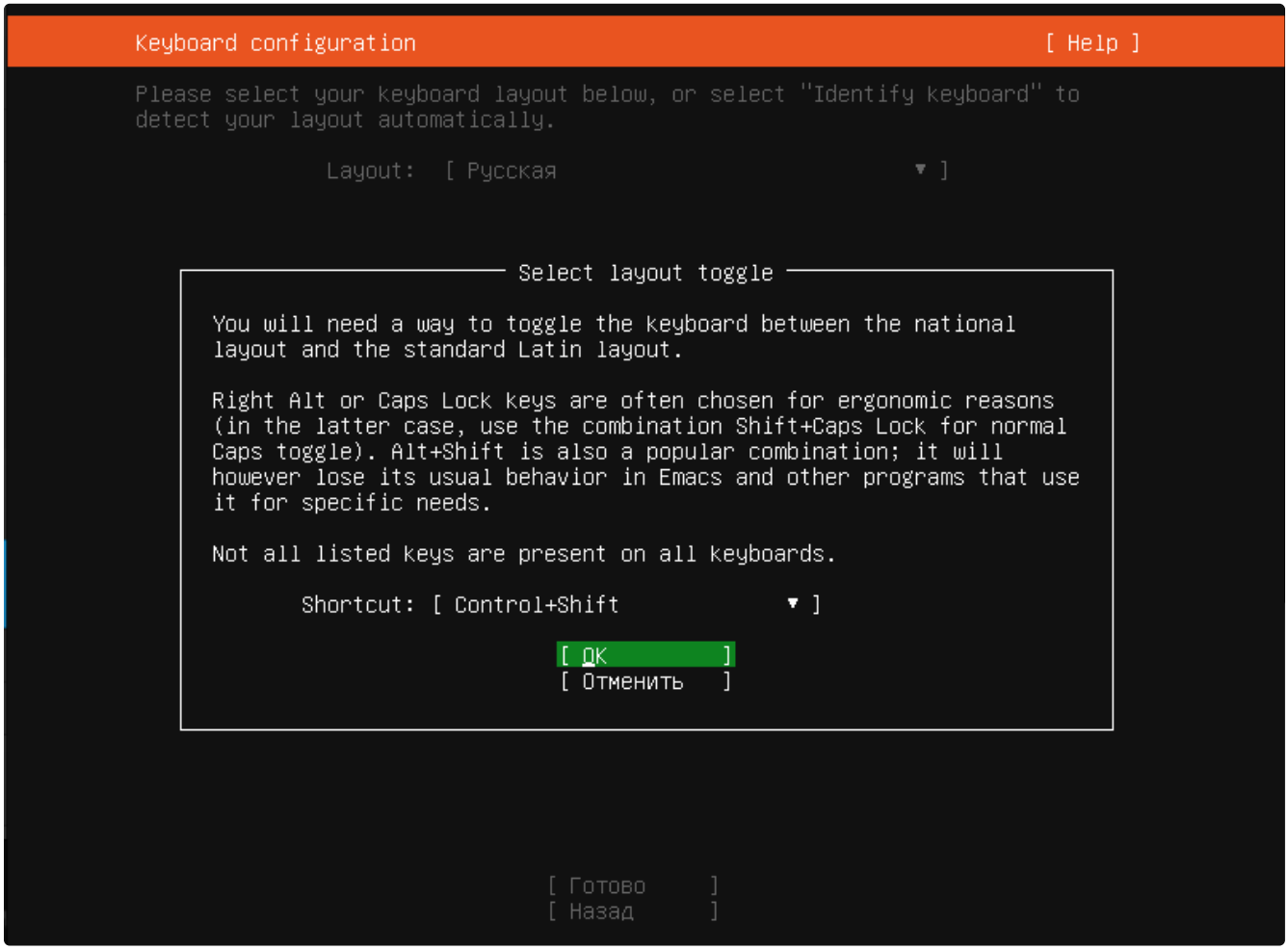
Продолжить без обновления



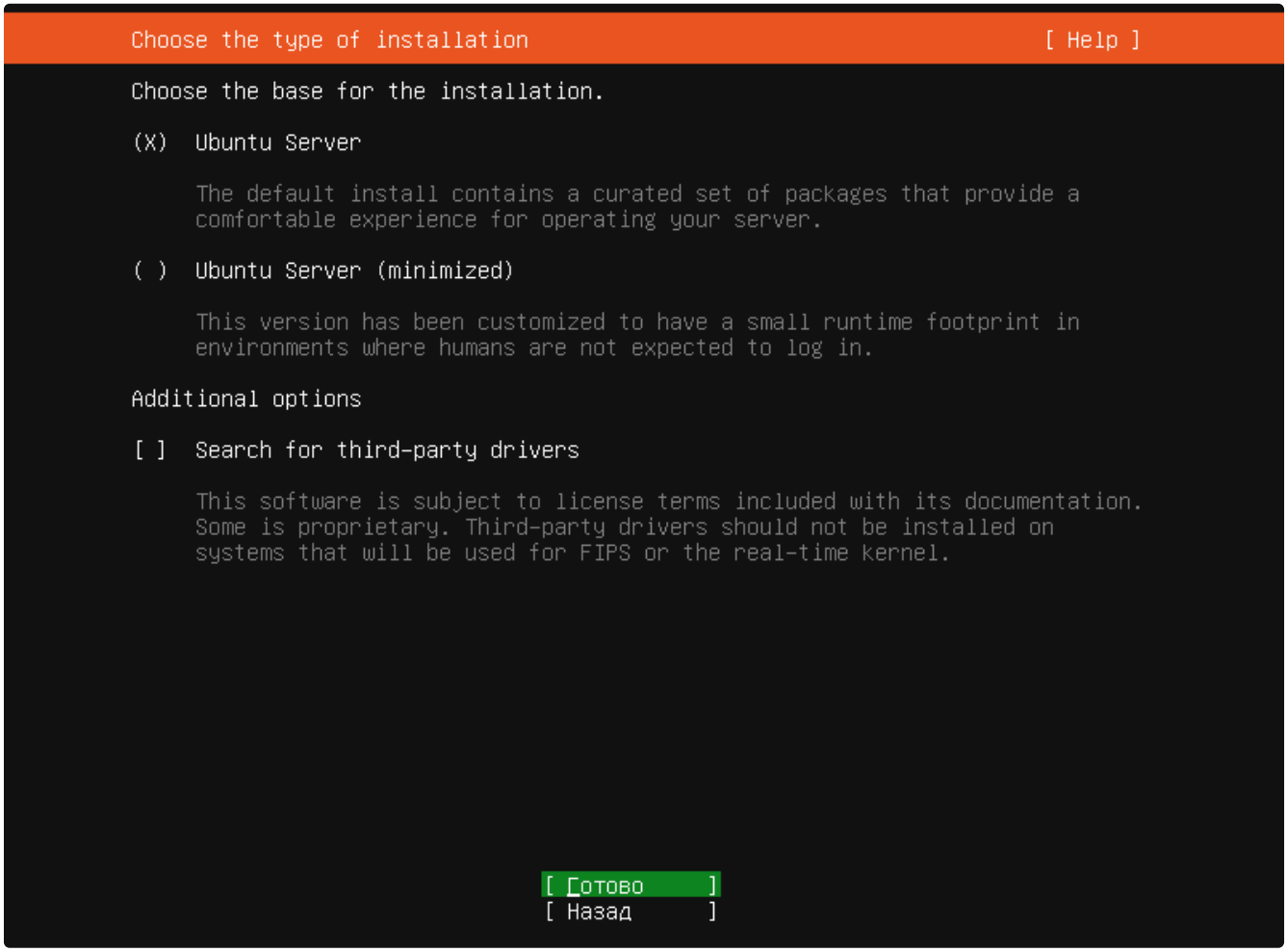
Выбираем раскладку клавиатуры



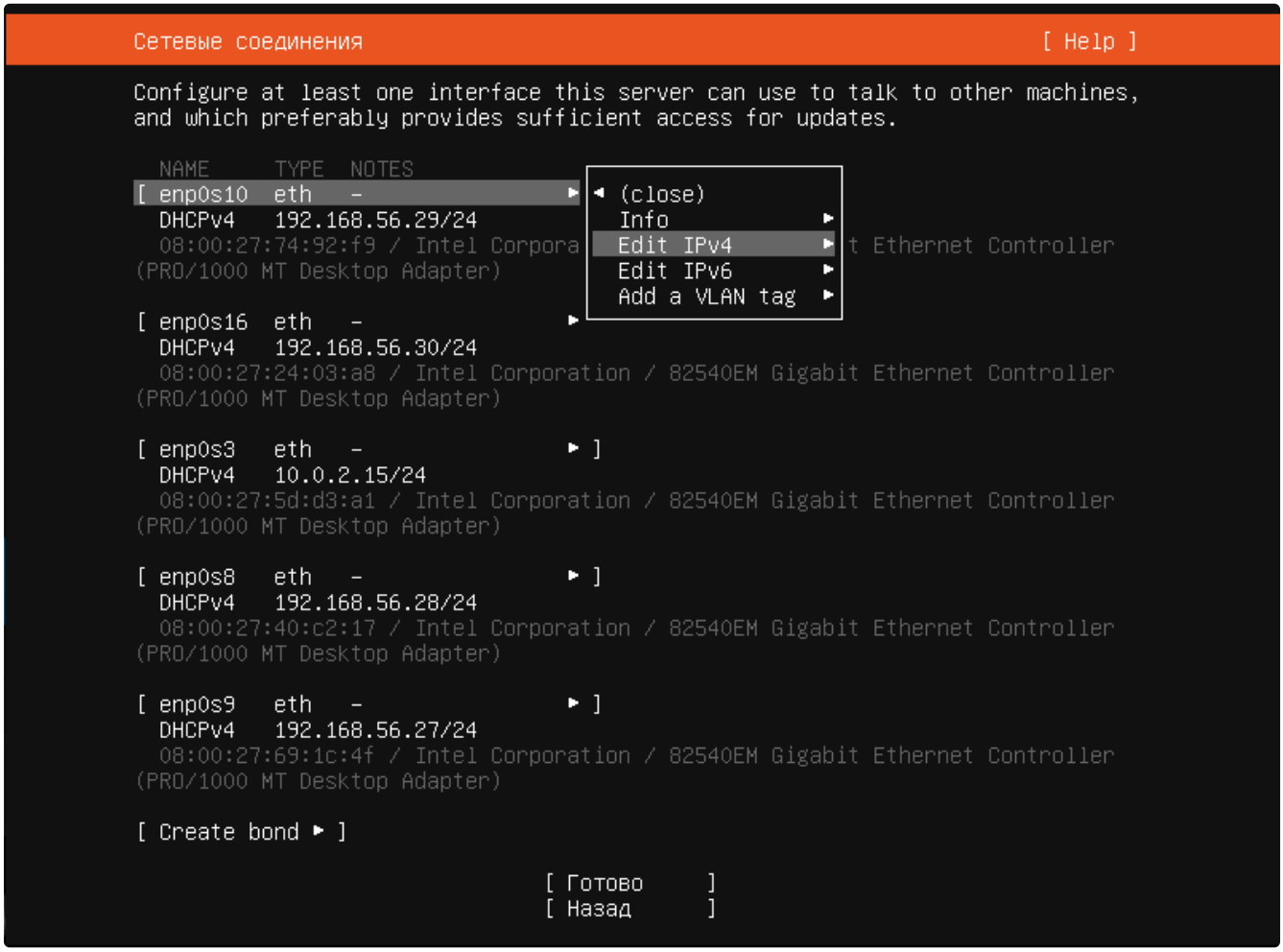
Выбираем переключатель раскладок Рус/Eng



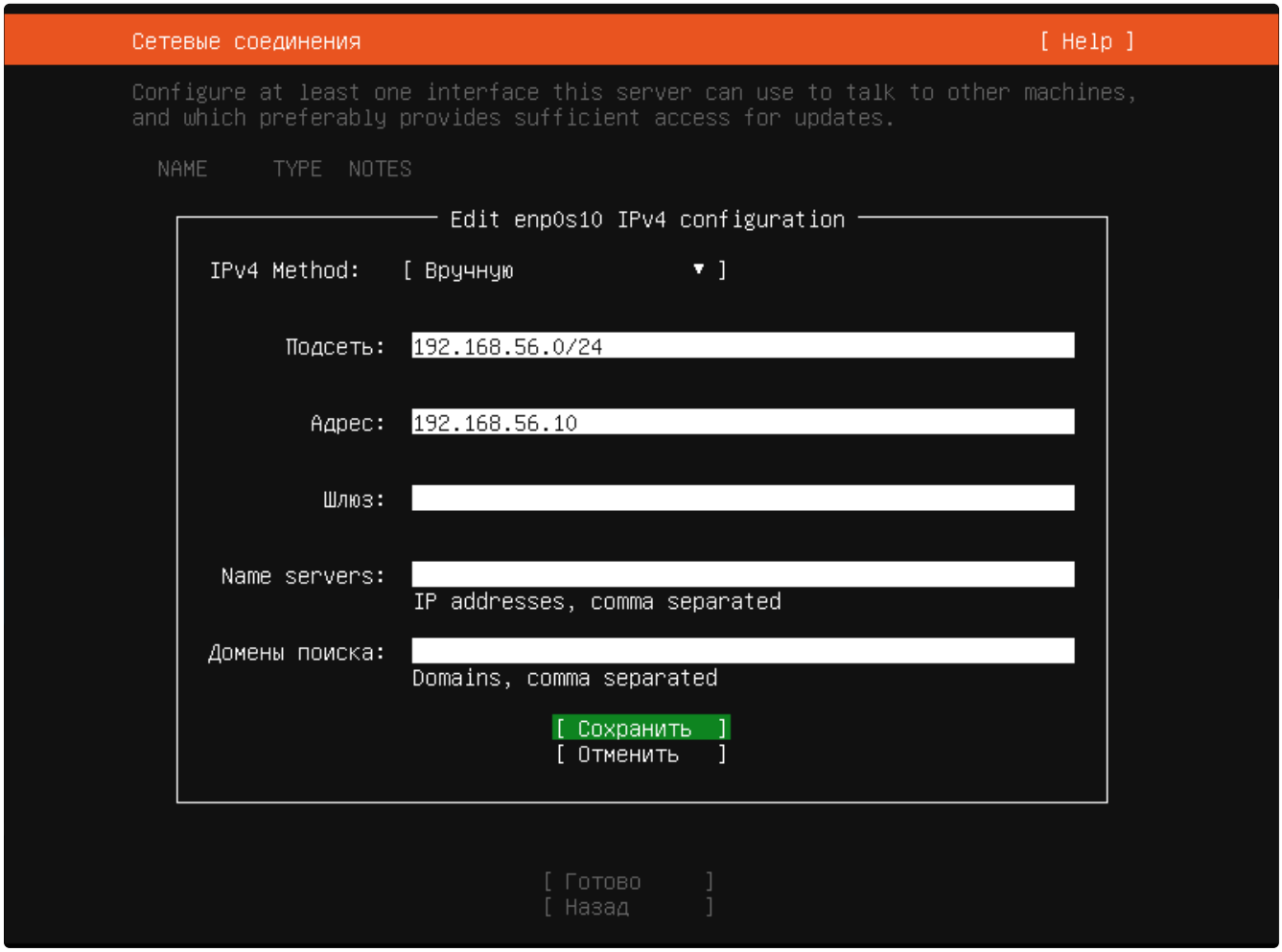
Выбираем стандартную установку Ubuntu Сервер



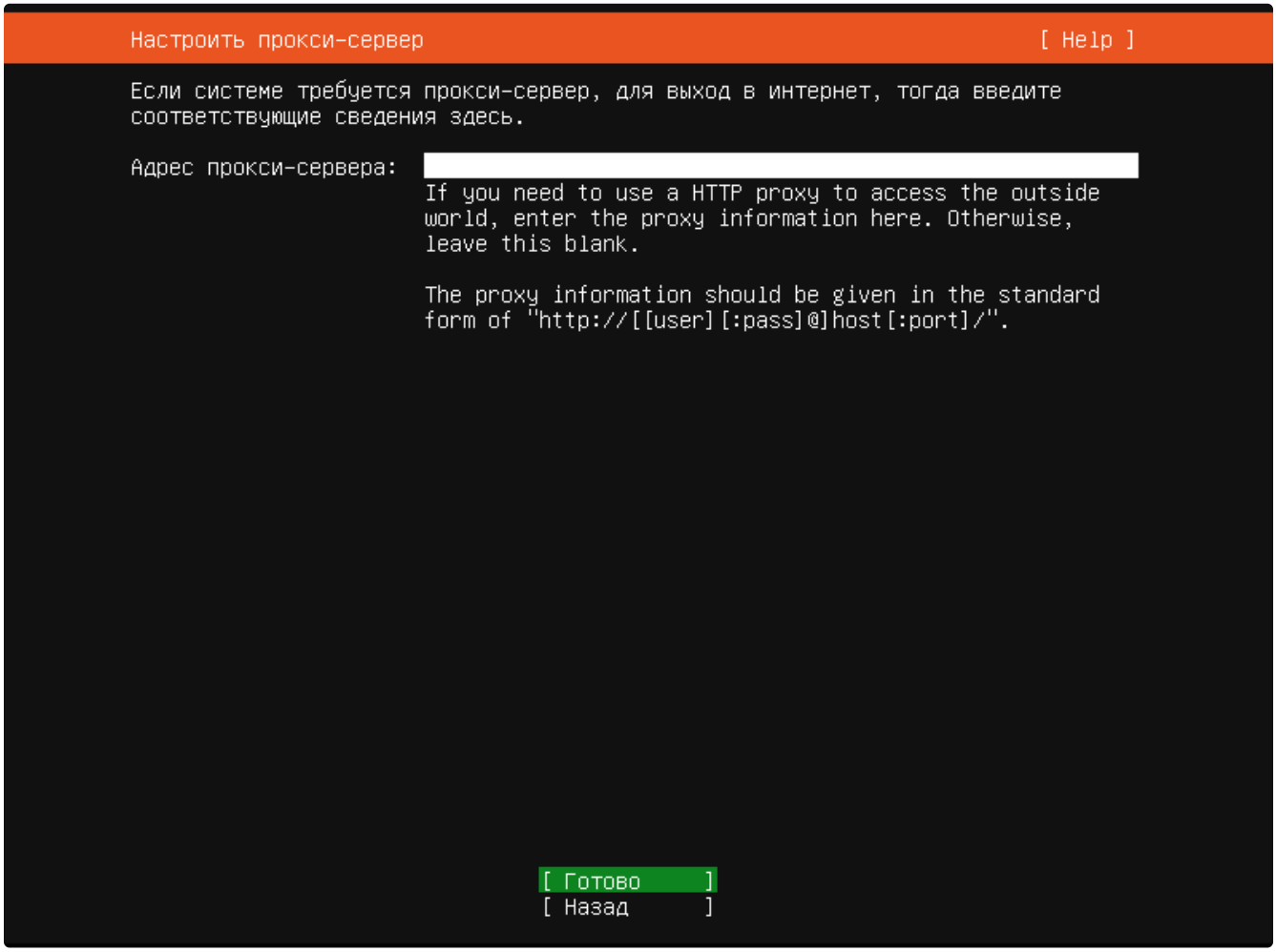
Выбираем Eht: интерфейс для установки Ubuntu (должен будет иметь доступ в интернет)



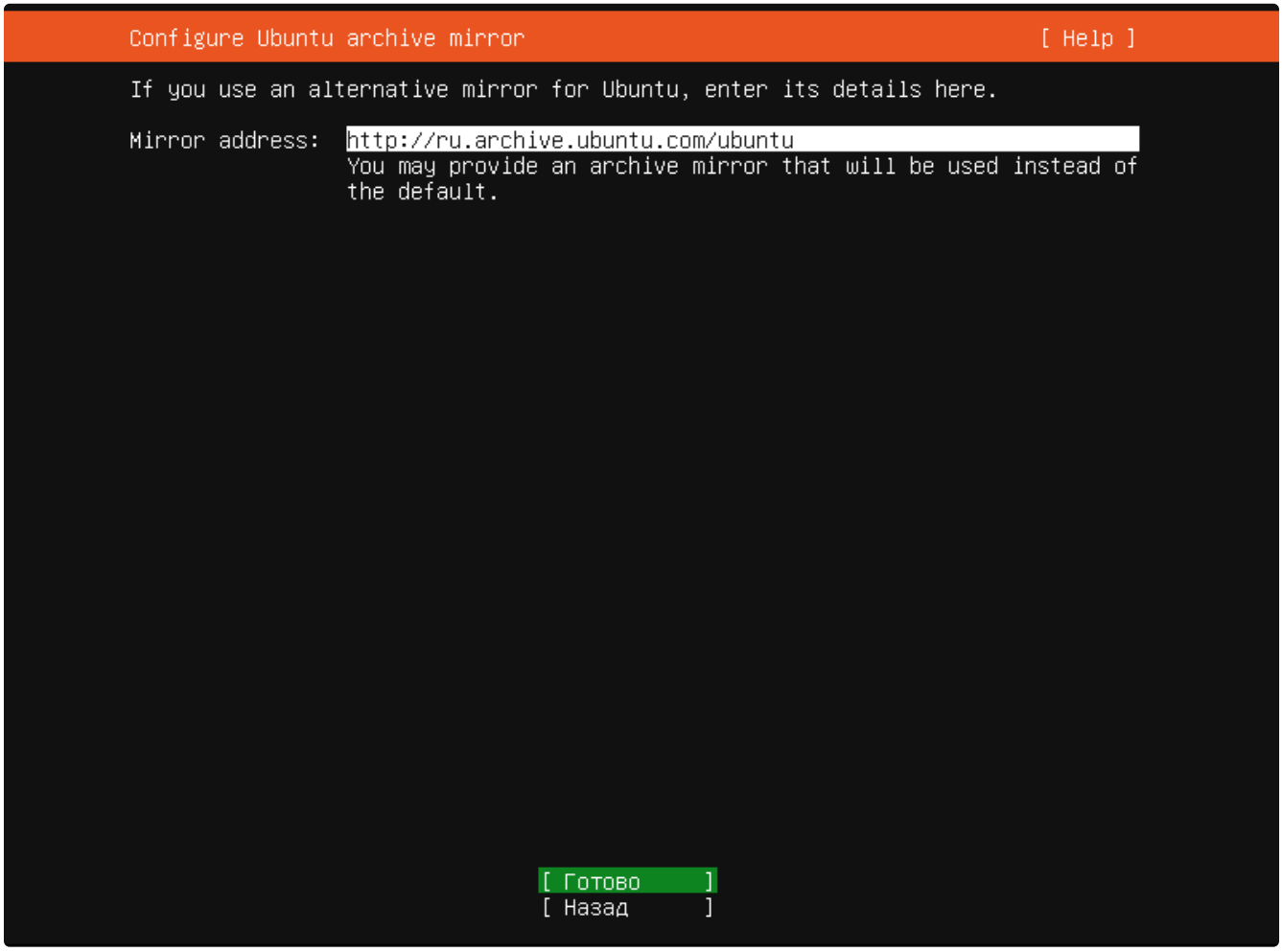
Установить IP-адрес для доступа в интернет(DHCP использовать нельзя)



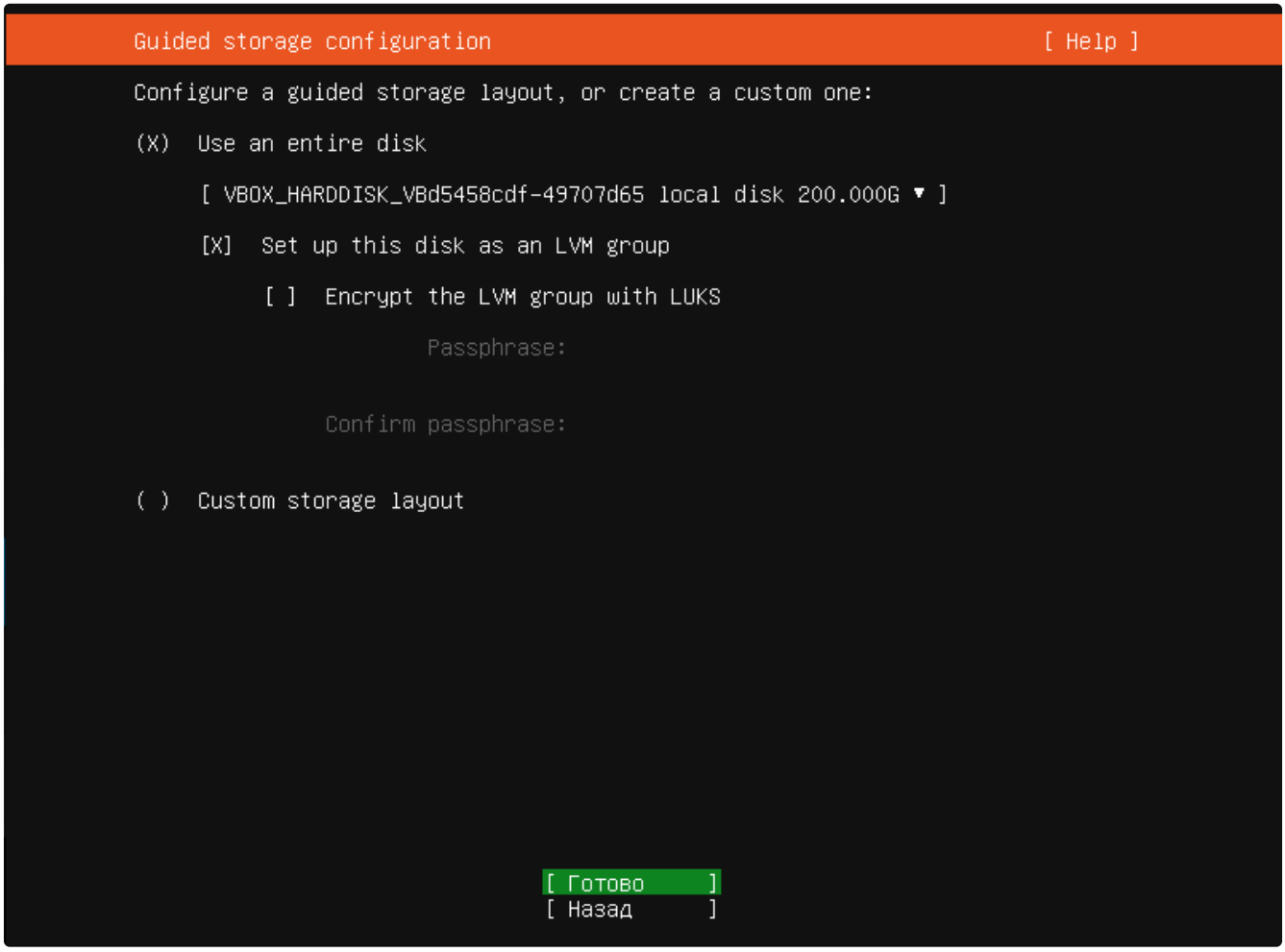
Прокси-сервер не используем(если в вашем проекте применяется то укажите)



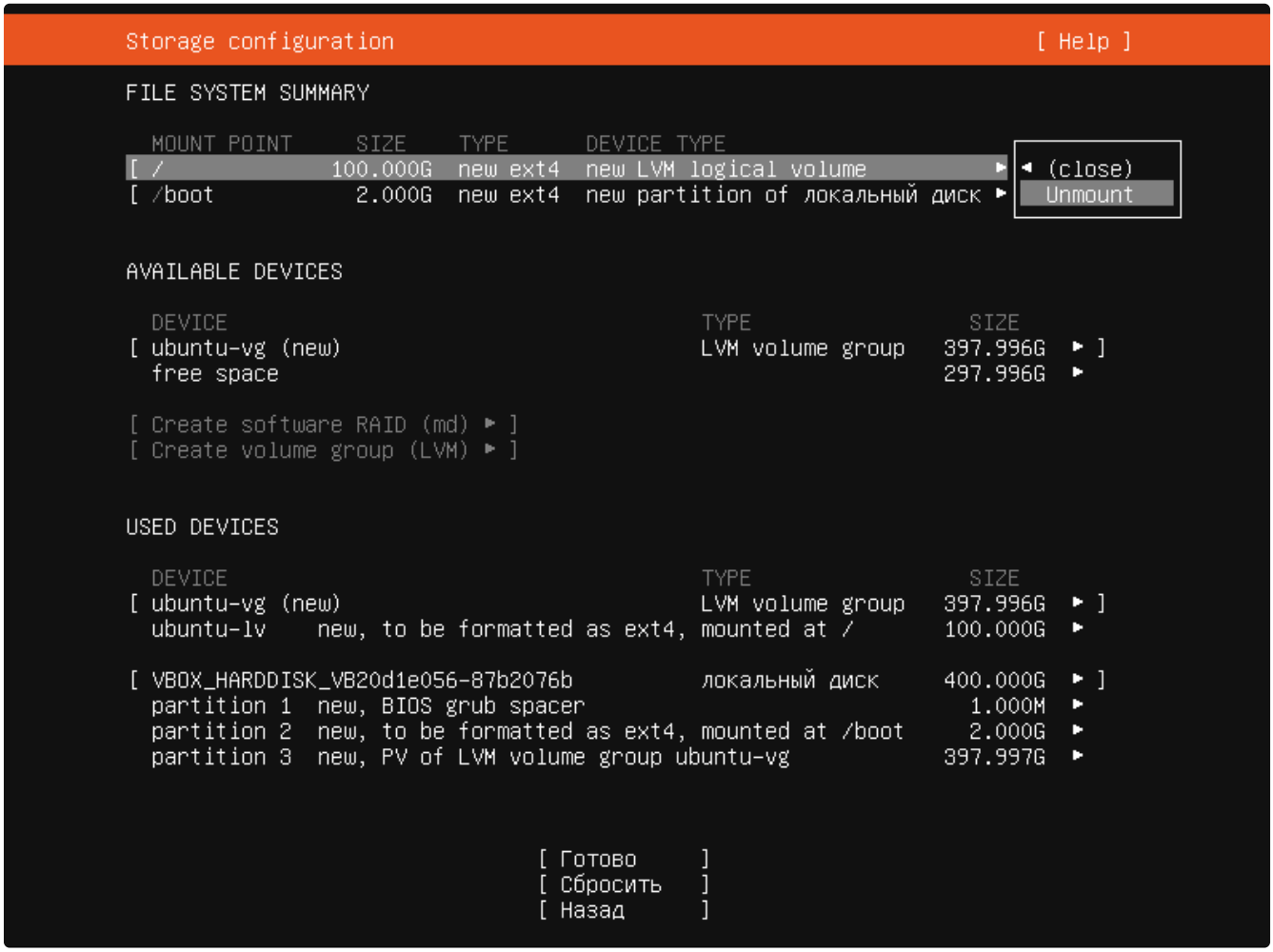
Значение по умолчанию



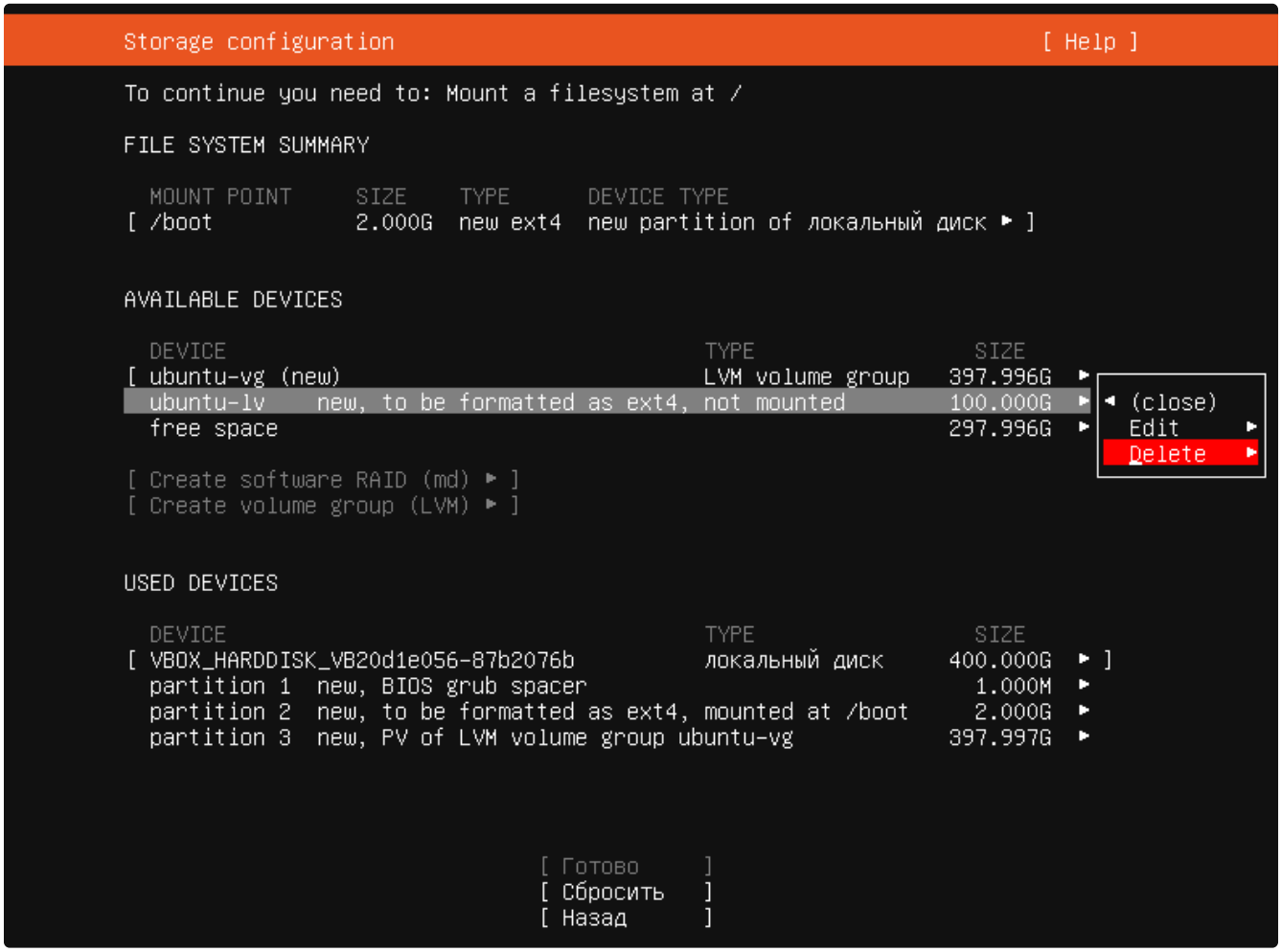
Создаем партиции на диске , используем LVM



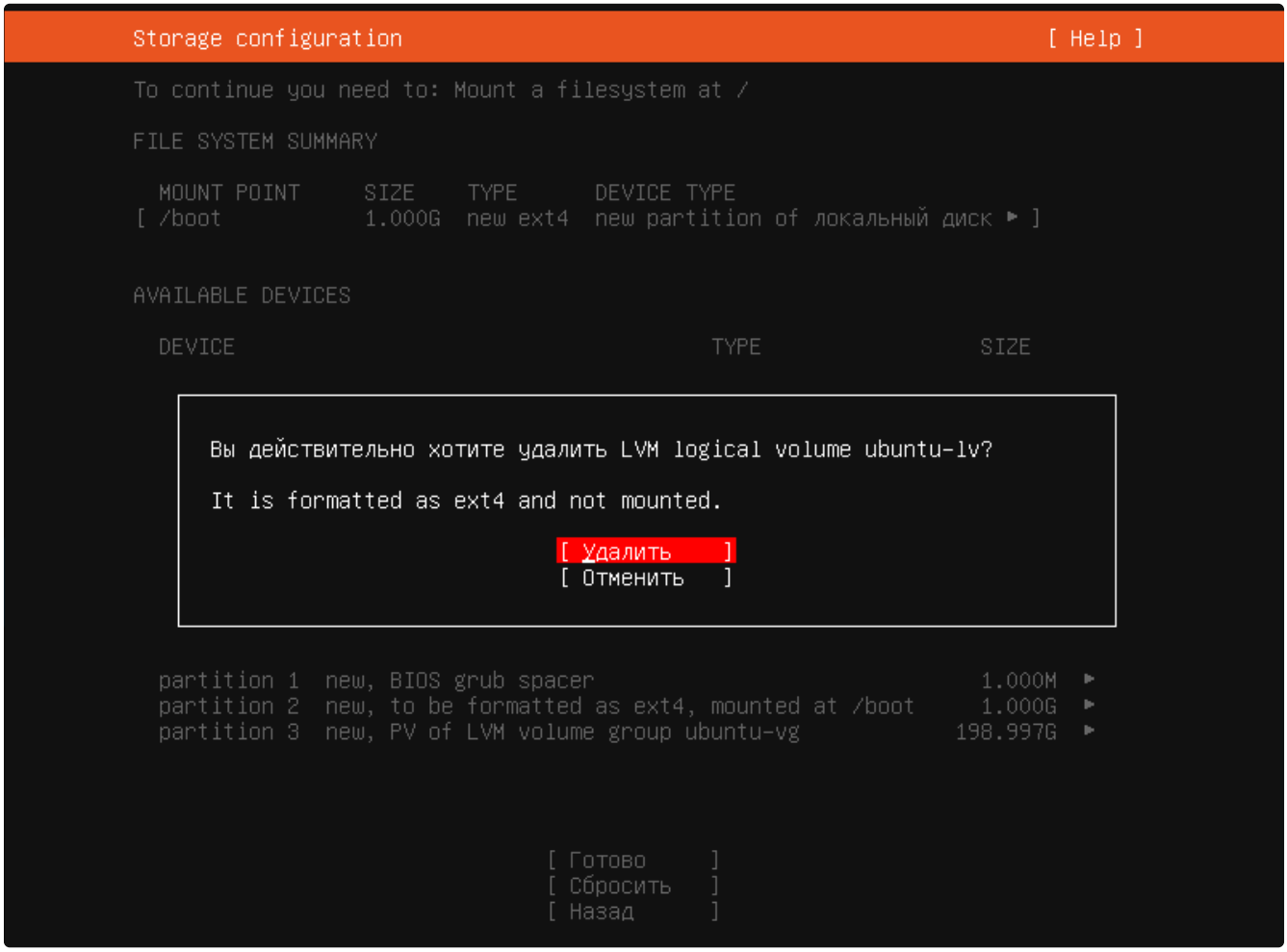
По умолчанию созданы две партиции /boot и / , /boot оставляем , а / - удаляем. Вначале размонтировать



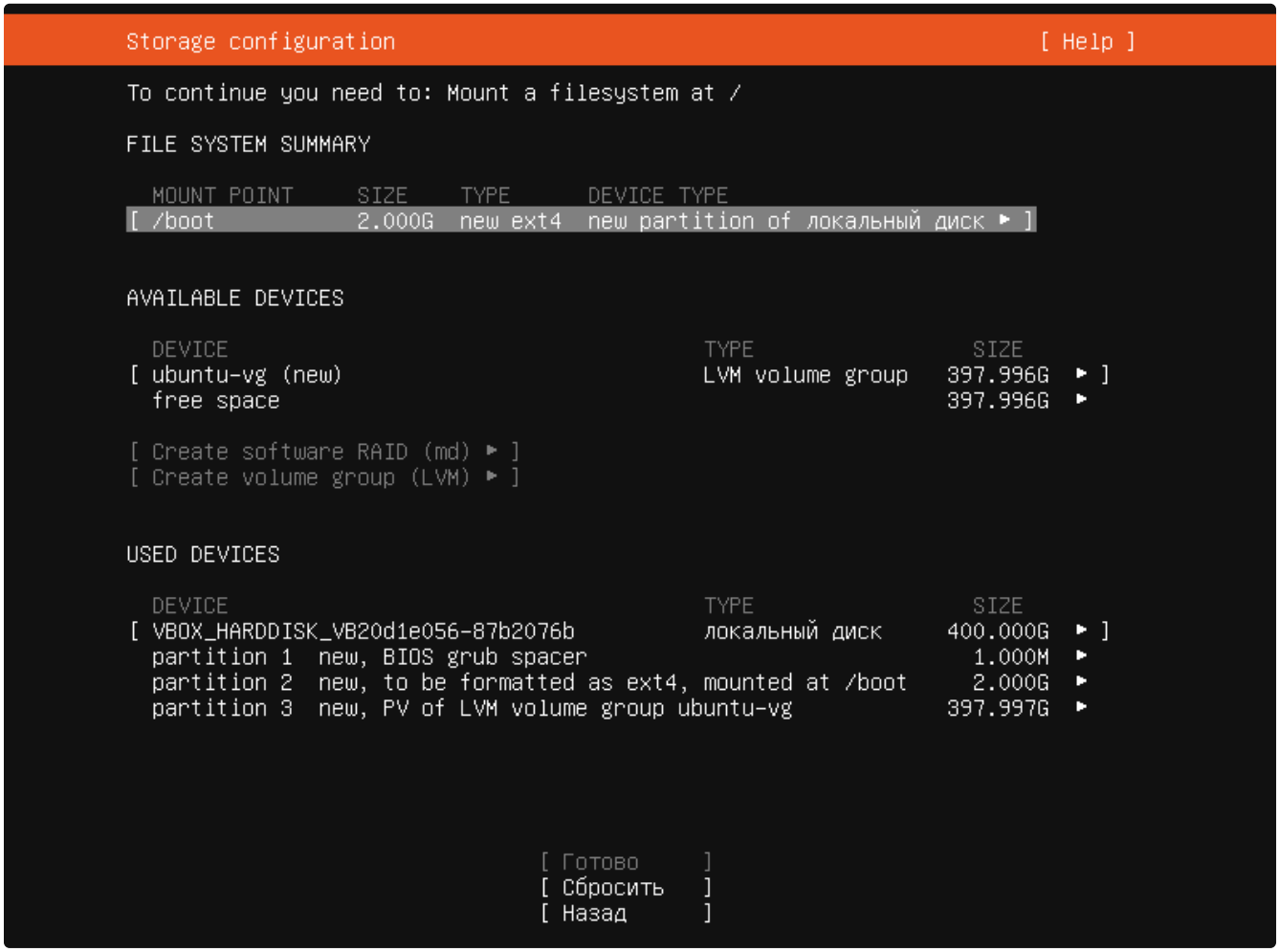
Затем удалить.



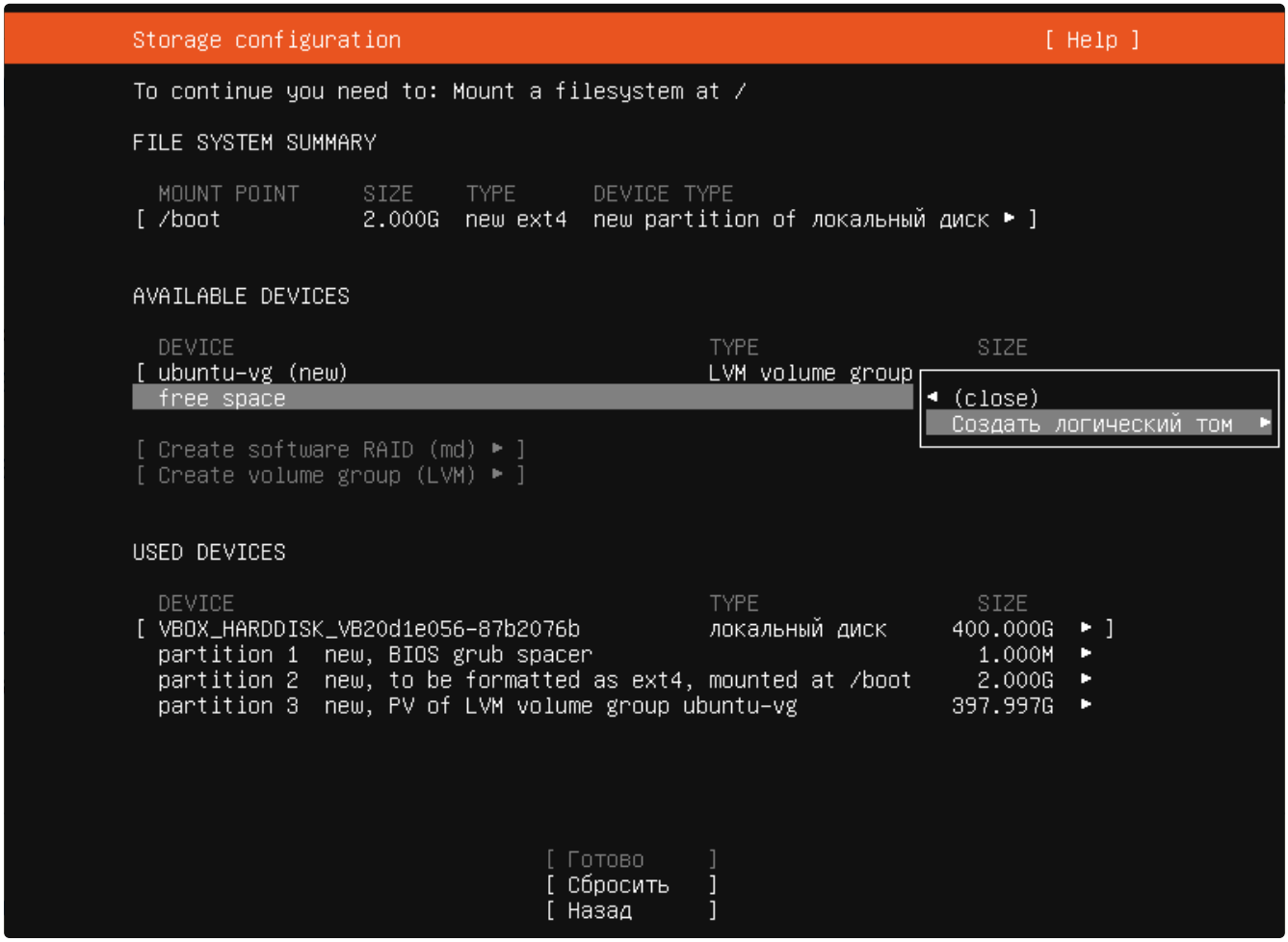
Подтвердить



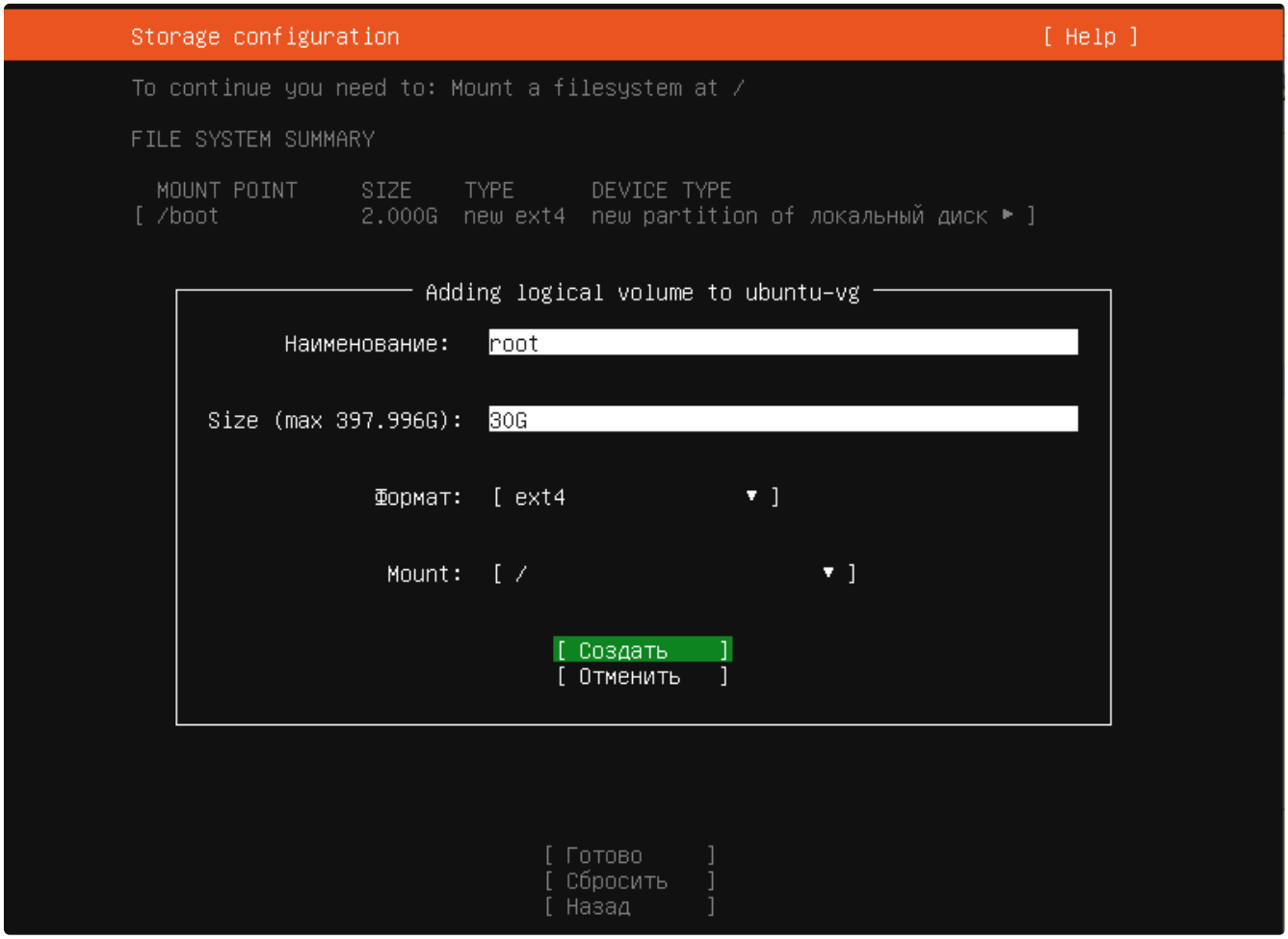
Оставить только /boot



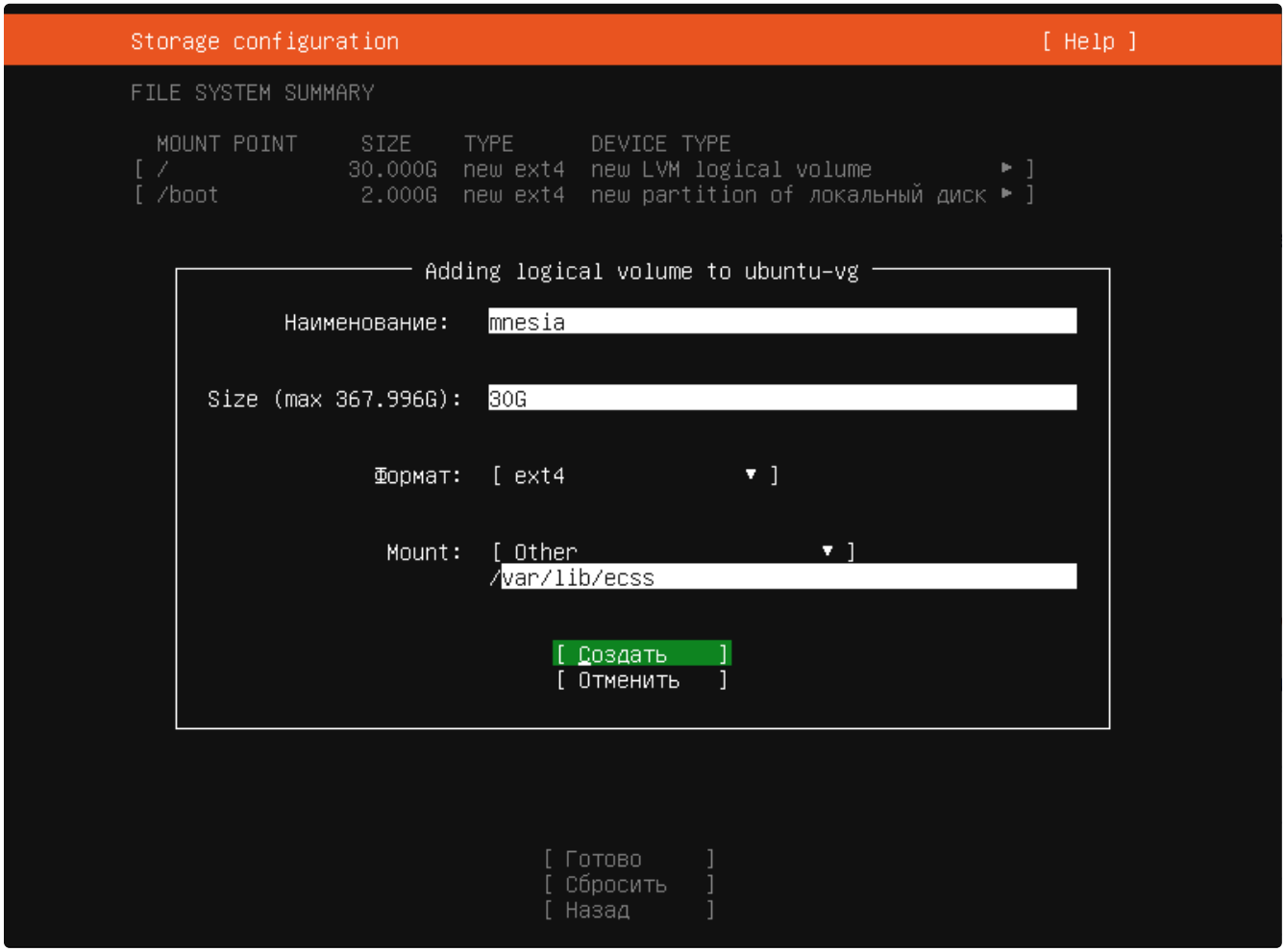
Создать партиции в соответствии с размерностью указанной в таблице ниже



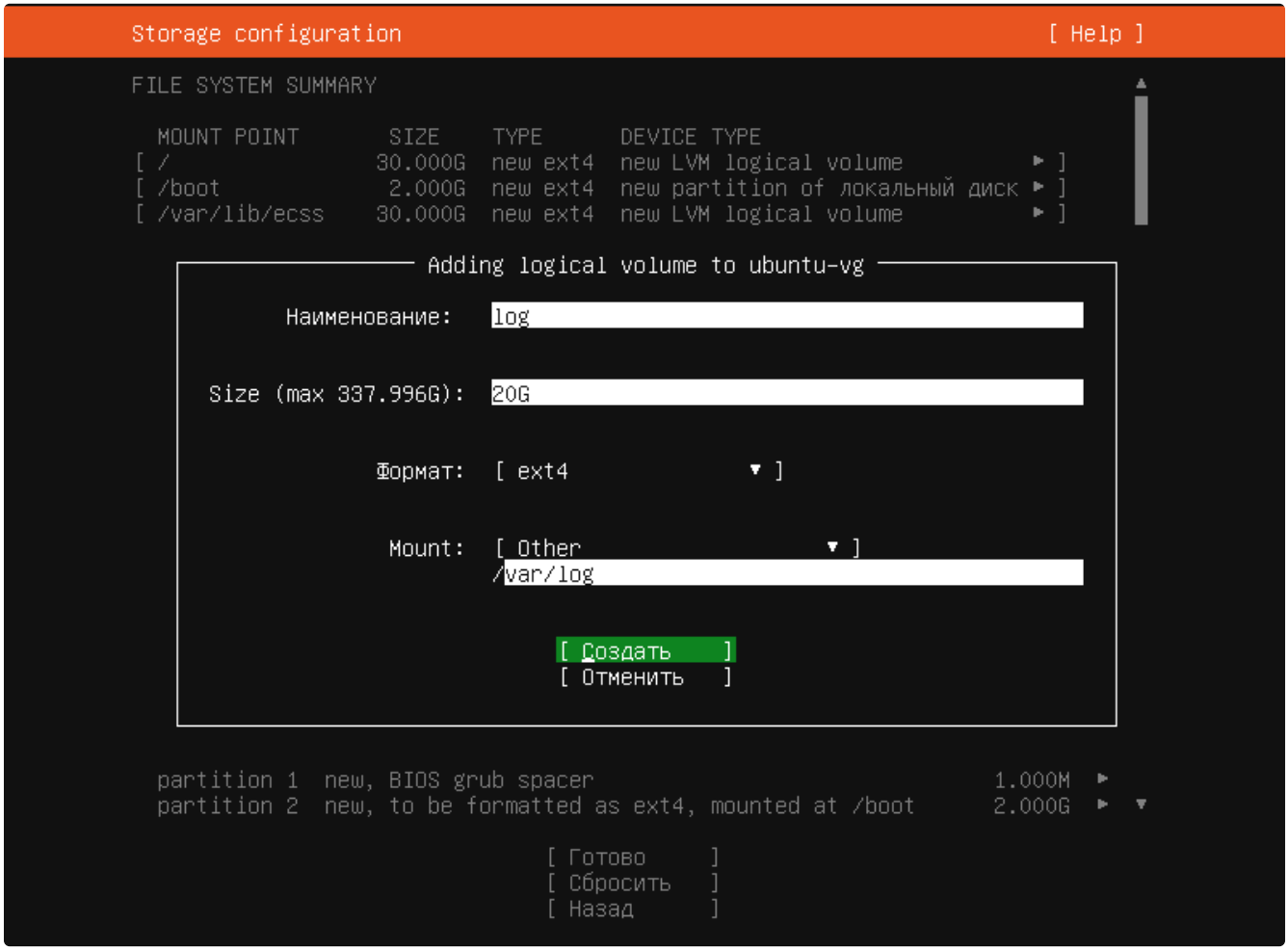
root размер 30 Гб , точка монтирования /



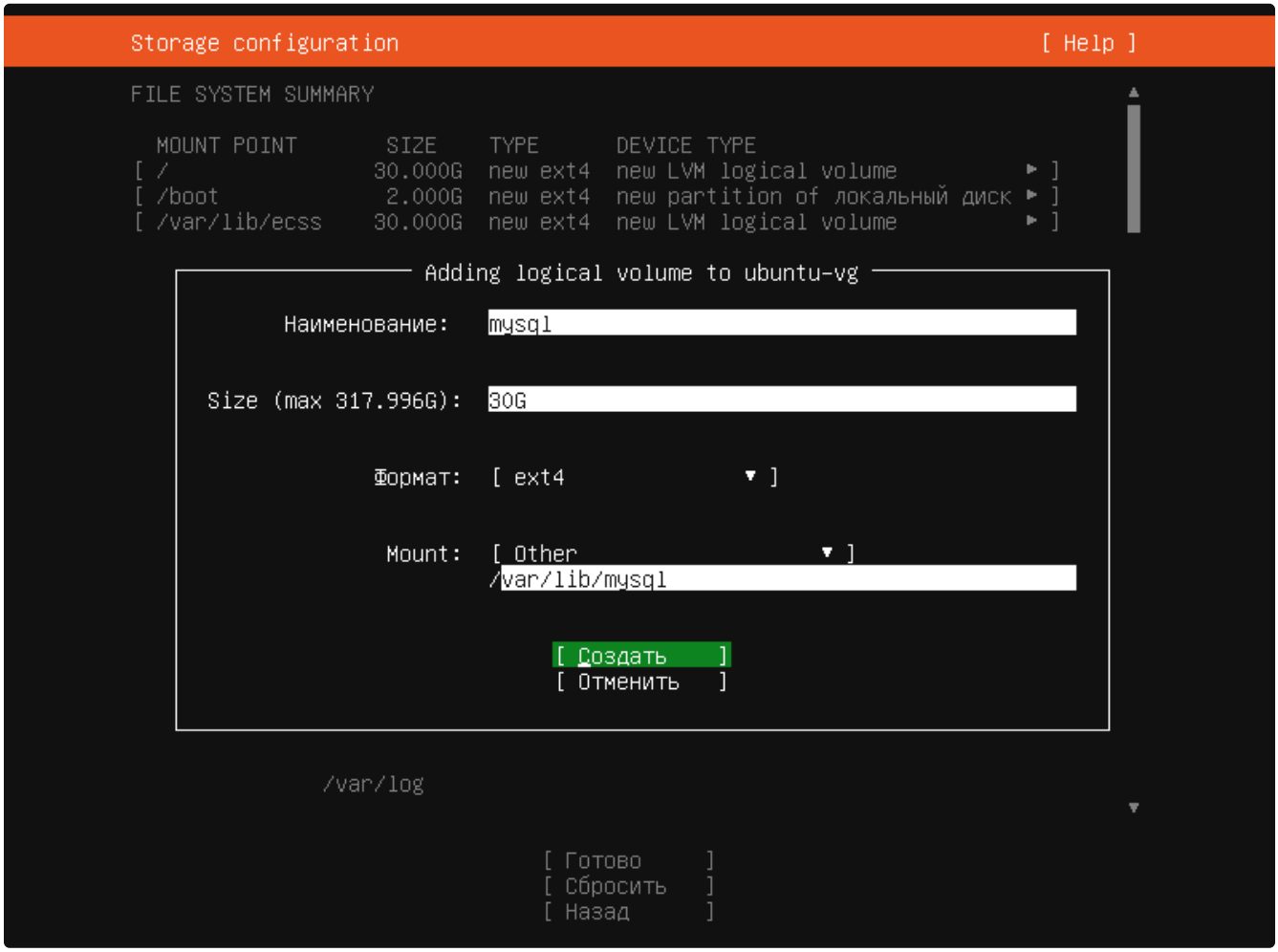
mnesia размер 30 Гб, точка монтирования /var/lib/ecss



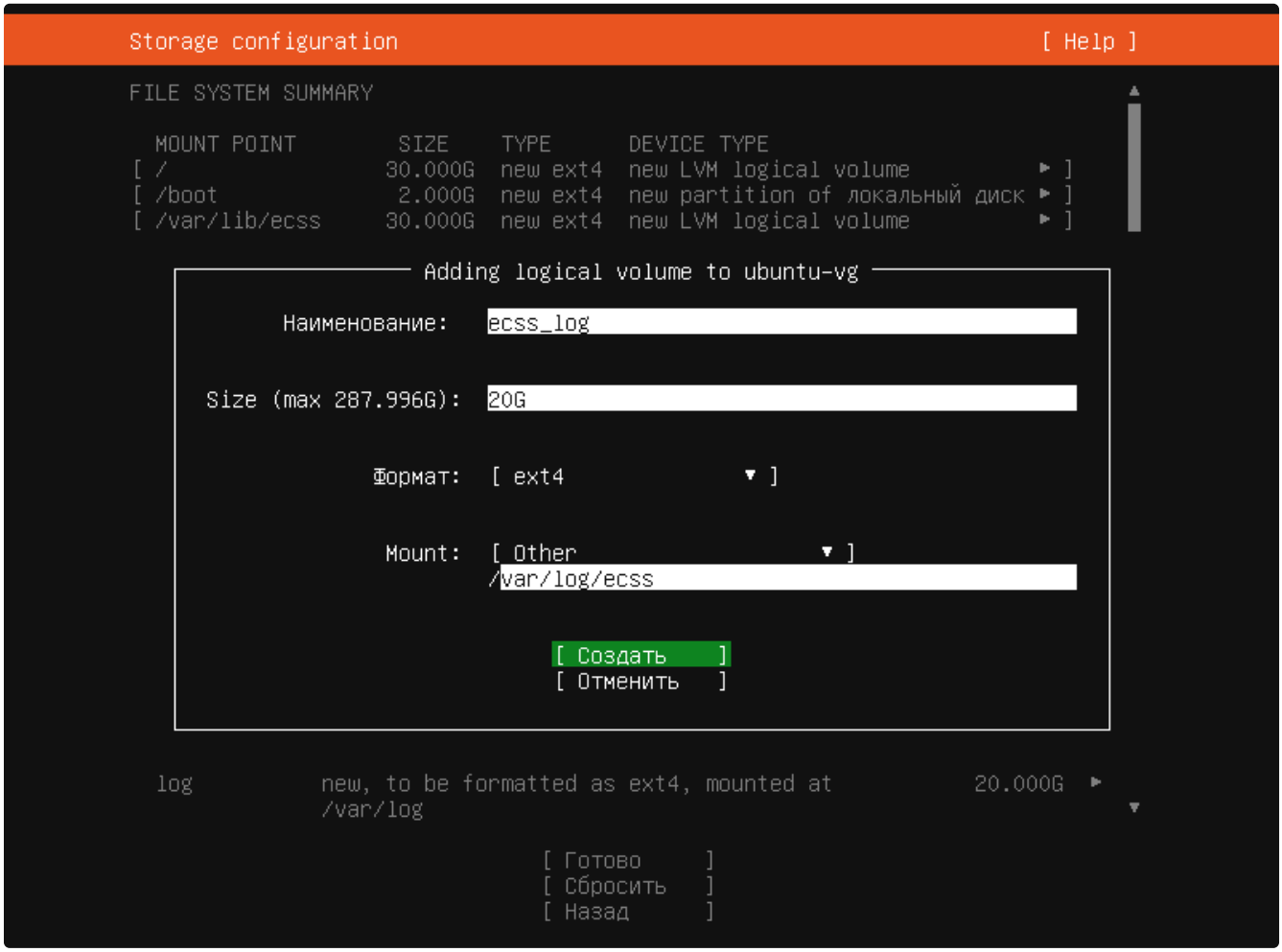
log размер 20 Гб, точка монтирования /var/log



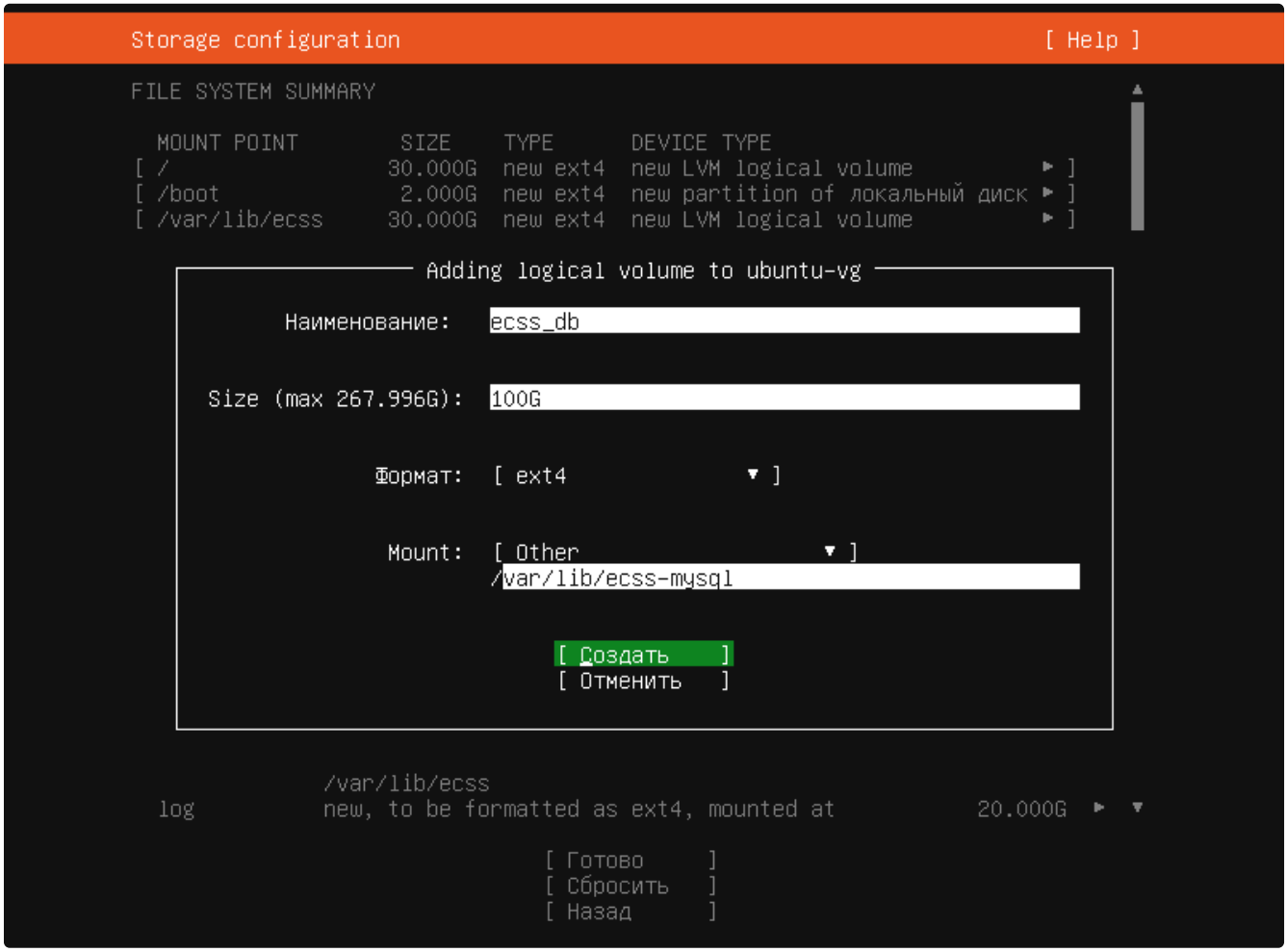
mysql, размер 30G, точка монтирования - /var/lib/mysql



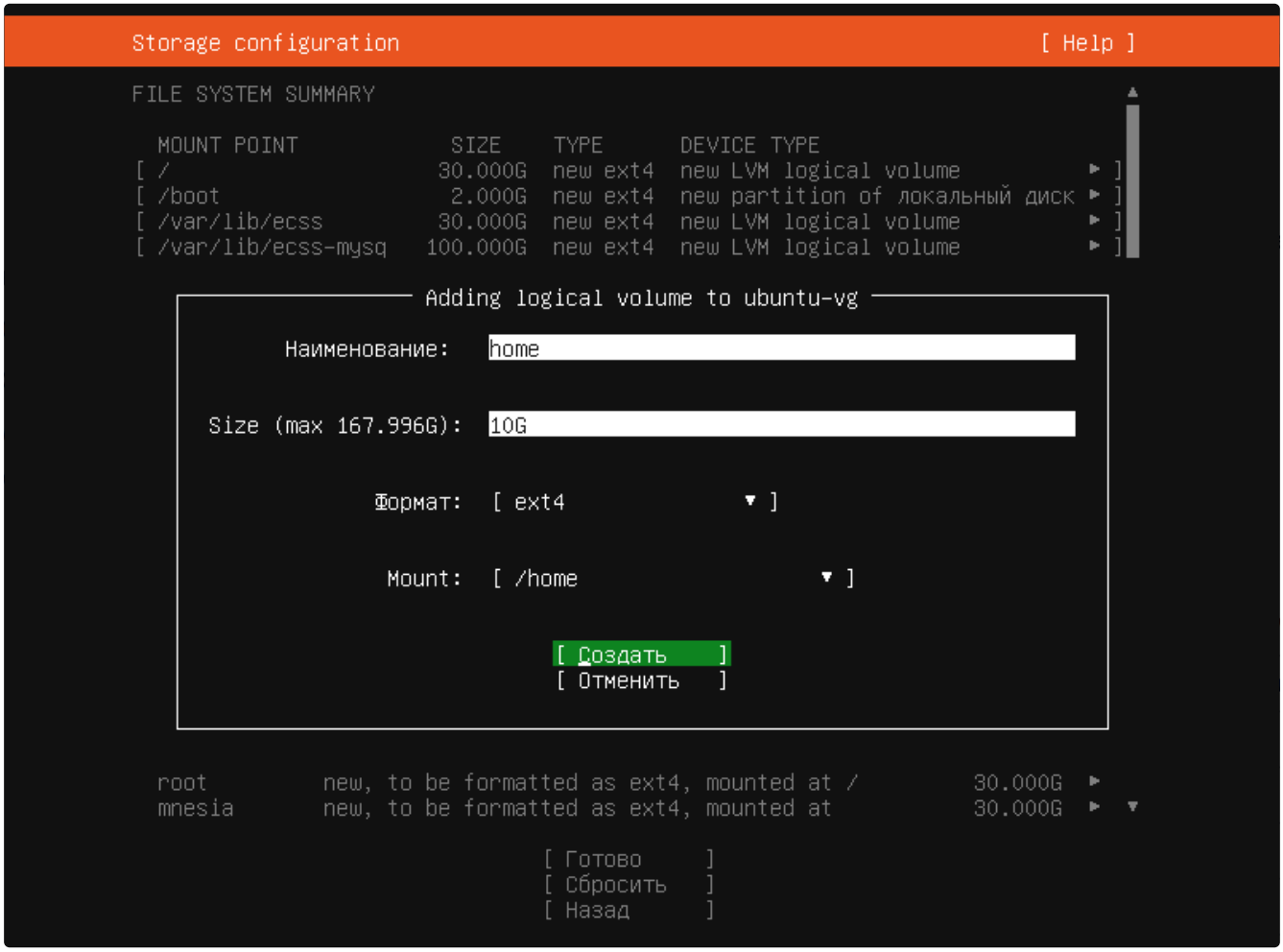
ecss_log, размер 20 Гб , точка монтирования /var/log/ecss



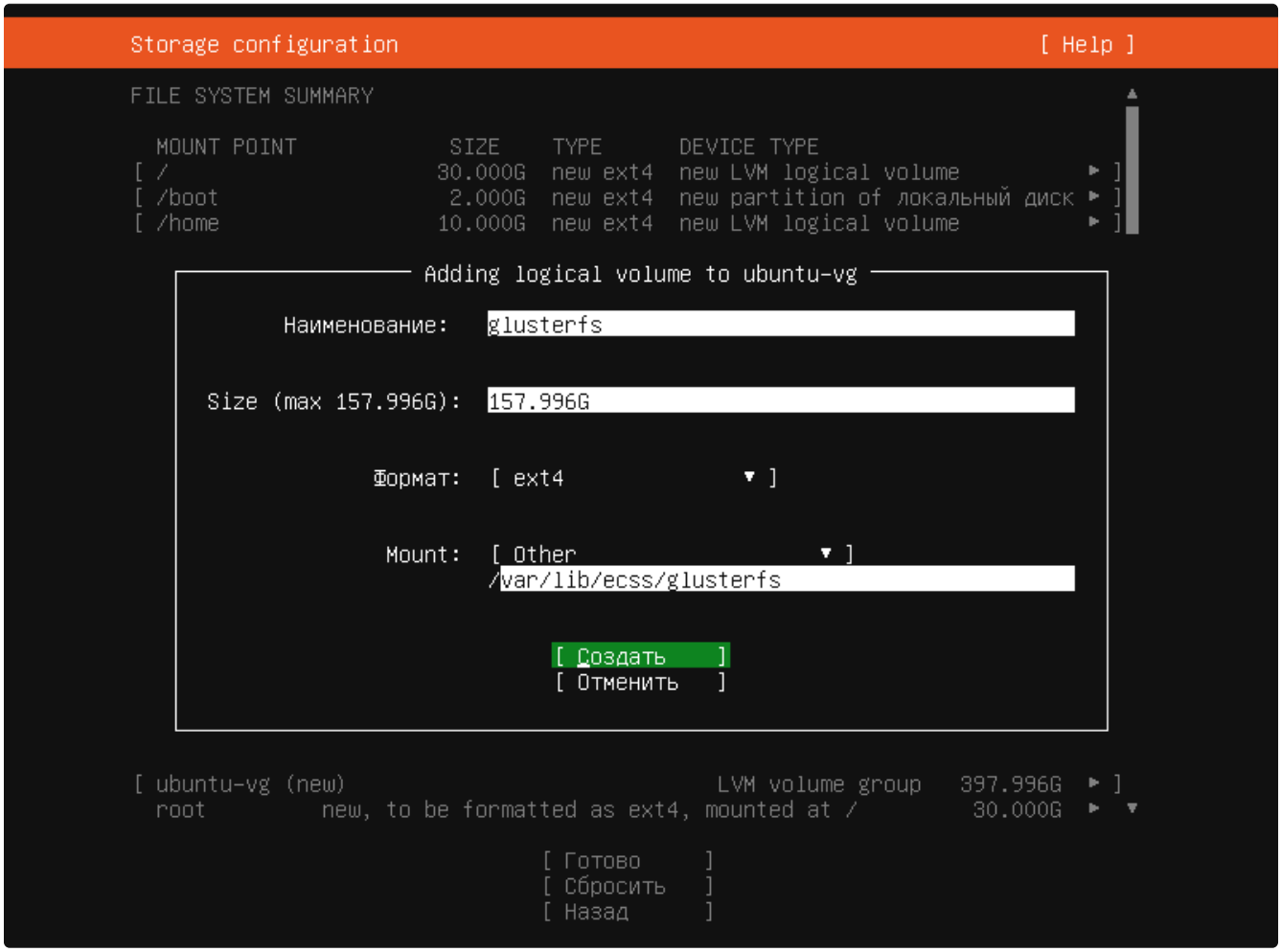
ecss_db, размер 100 Гб , точка монтирования - /var/lib/ecss-mysql



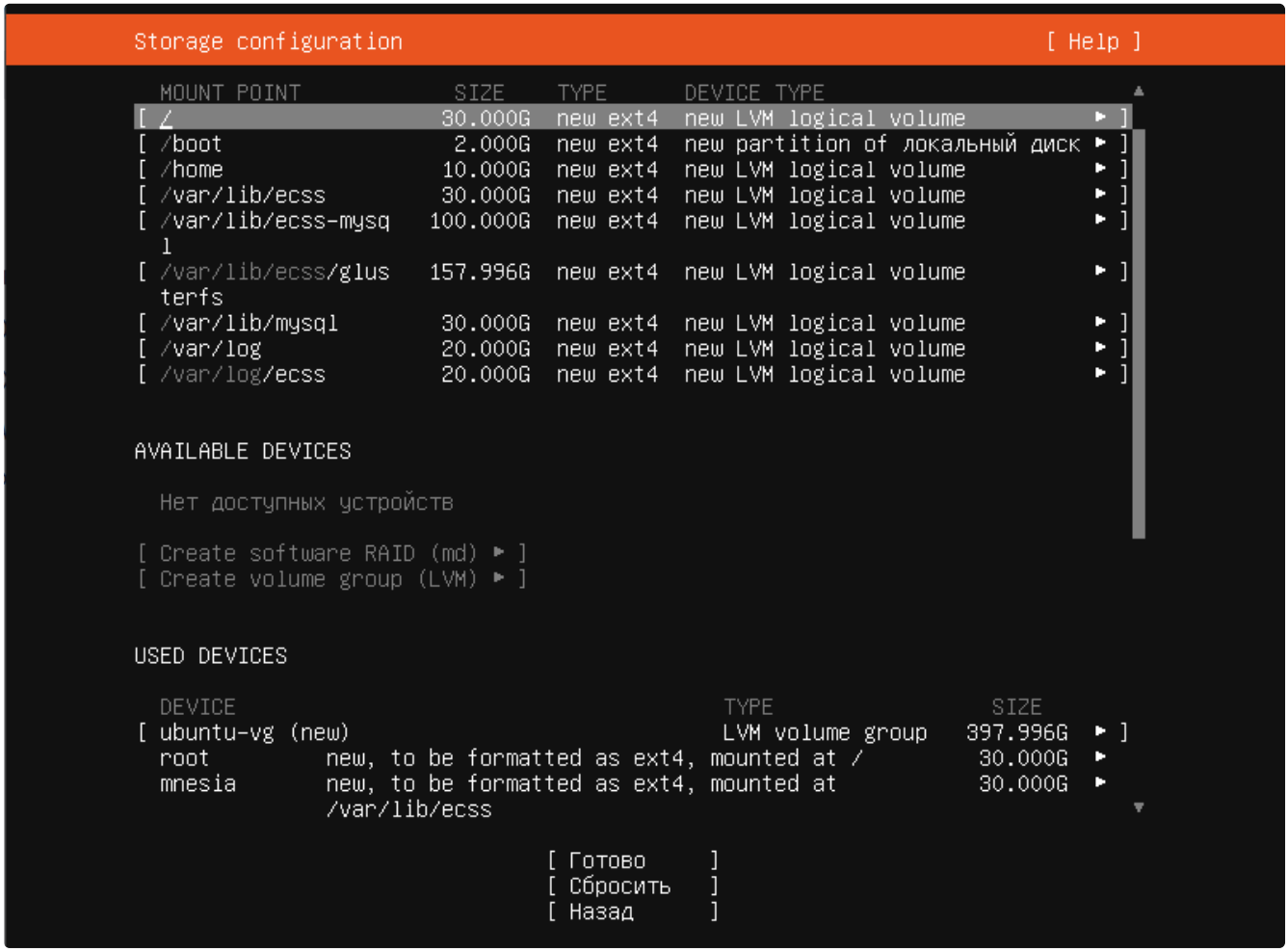
home, размер 10 Гб, точка монтирования - /home

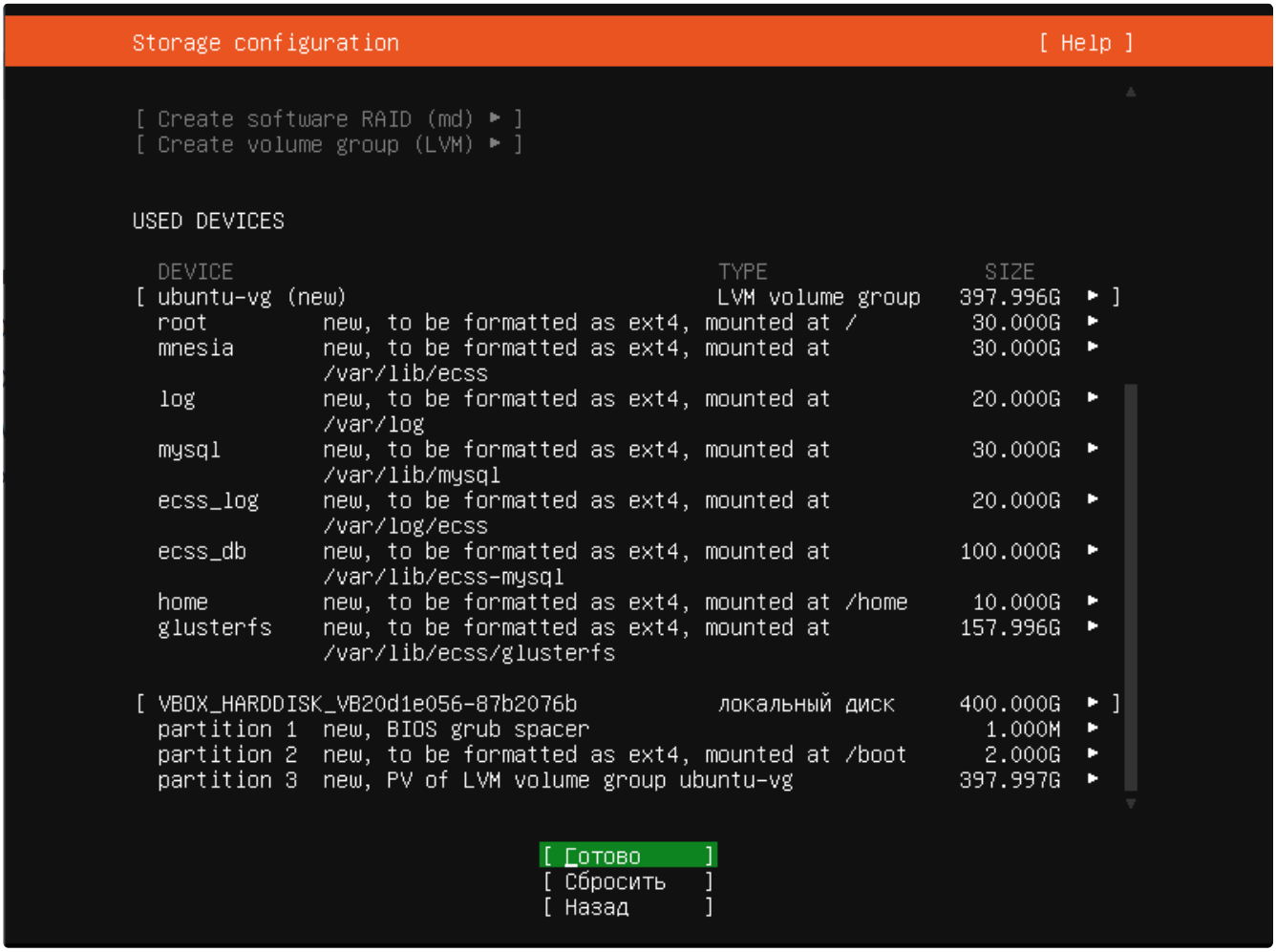


glusterfs - для кластера / для одного хоста - restfs , размер - все оставшееся место на диске, точка монтирования - /var/lib/ecss/glusterfs - для кластера , и /var/lib/ecss/restfs - для одного сервера

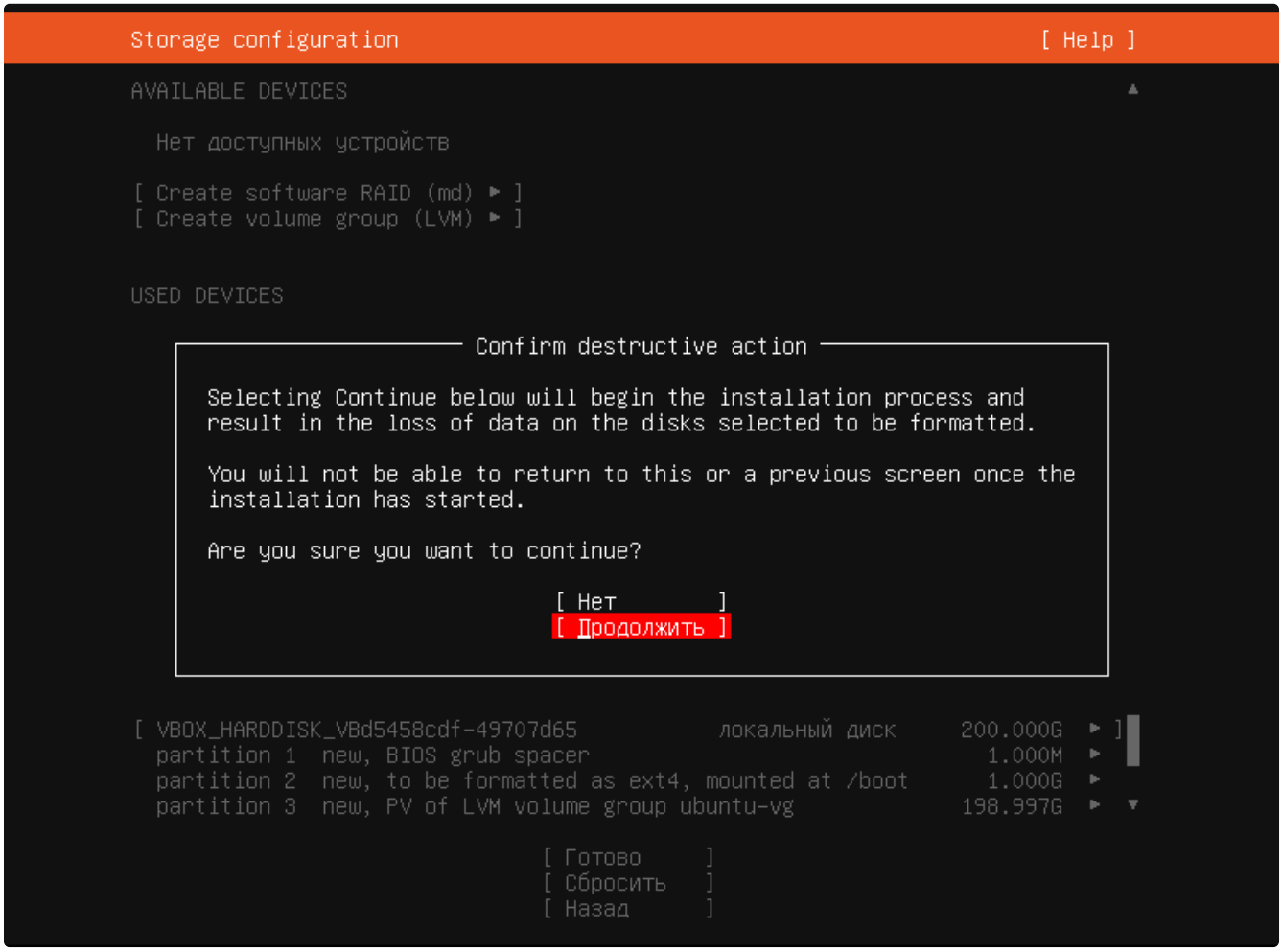


полученная конфигурация

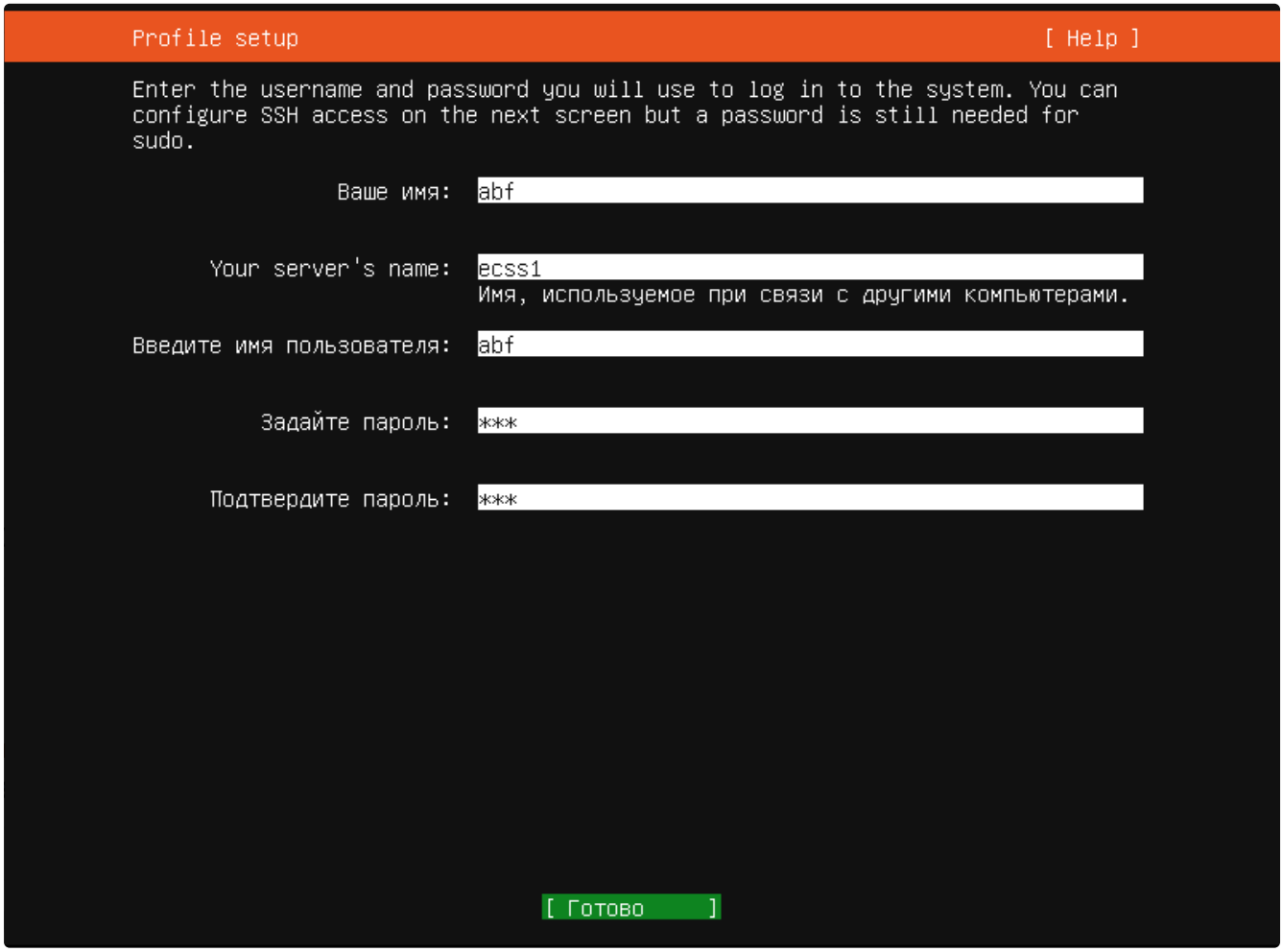

Storage configuration
[Help]
MOUNT POINT SIZE TYPE DEVICE TYPE
[/ 30.000G new ext4 new LVM logical volume ▸]
[/boot 2.000G new ext4 new partition of локальный диск ▸]
[/home 10.000G new ext4 new LVM logical volume ▸]
[/var/lib/ecss 30.000G new ext4 new LVM logical volume ▸]
[/var/lib/ecss-mysq 100.000G new ext4 new LVM logical volume ▸]
l
[/var/lib/ecss/glus 157.996G new ext4 new LVM logical volume ▸]
terfs
[/var/lib/mysql 30.000G new ext4 new LVM logical volume ▸]
[/var/log 20.000G new ext4 new LVM logical volume ▸]
[/var/log/ecss 20.000G new ext4 new LVM logical volume ▸]
AVAILABLE DEVICES
Нет доступных устройств
[Create software RAID (md) ▸]
[Create volume group (LVM) ▸]
USED DEVICES
DEVICE TYPE SIZE
[ubuntu-vg (new) LVM volume group 397.996G ▸]
root new, to be formatted as ext4, mounted at / 30.000G ▸
mnesia new, to be formatted as ext4, mounted at /var/lib/ecss 30.000G ▸
[Готово]
[Сбросить]
[Назад]



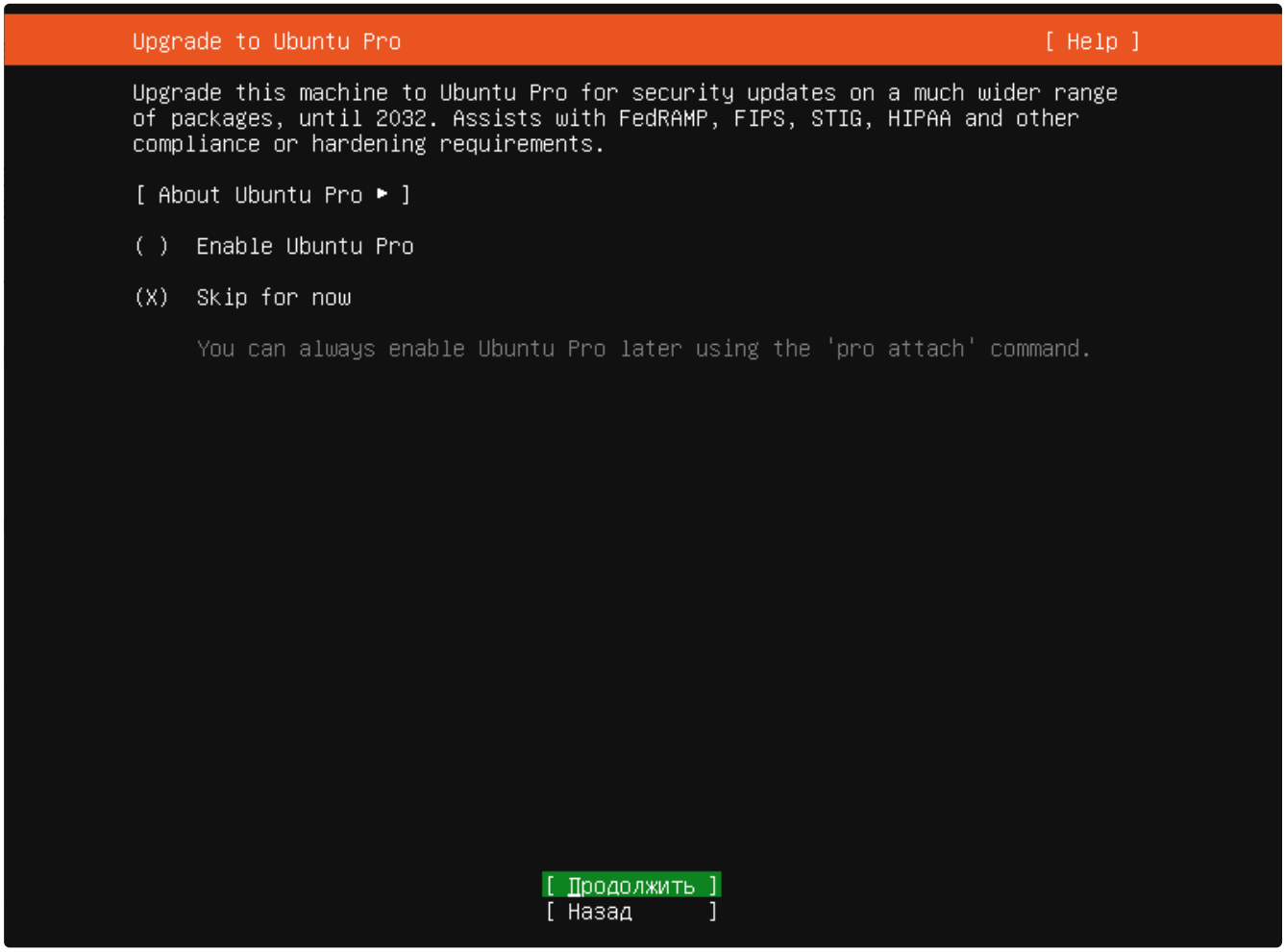
Подтвердить



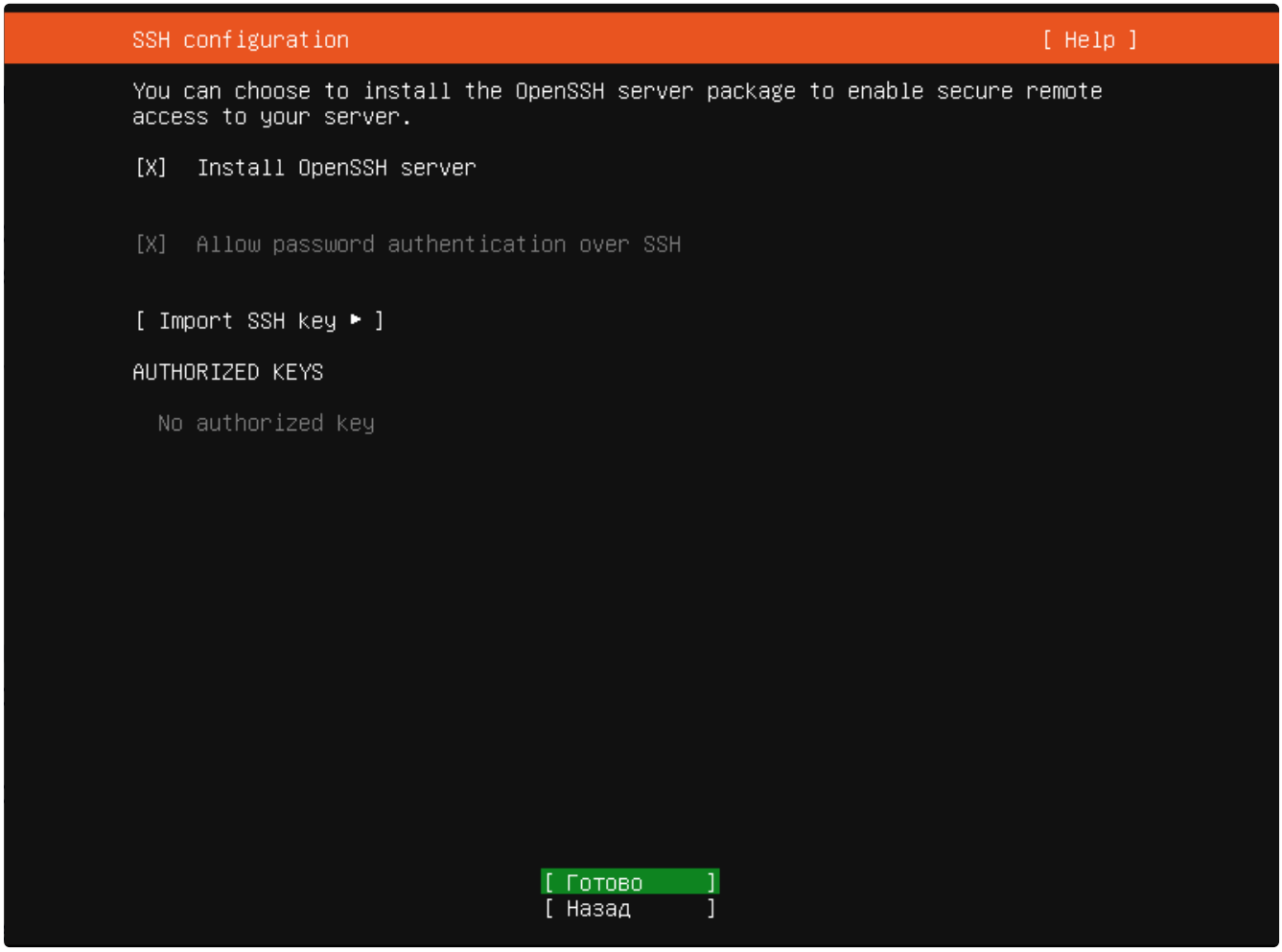
Создать пользователя и имя хоста, в ECSS10 рекомендуемое имя хоста ecss1/ecss2 (используется при интеграции лицензий), имя пользователя (в примере abf / пароль на ваше усмотрение)



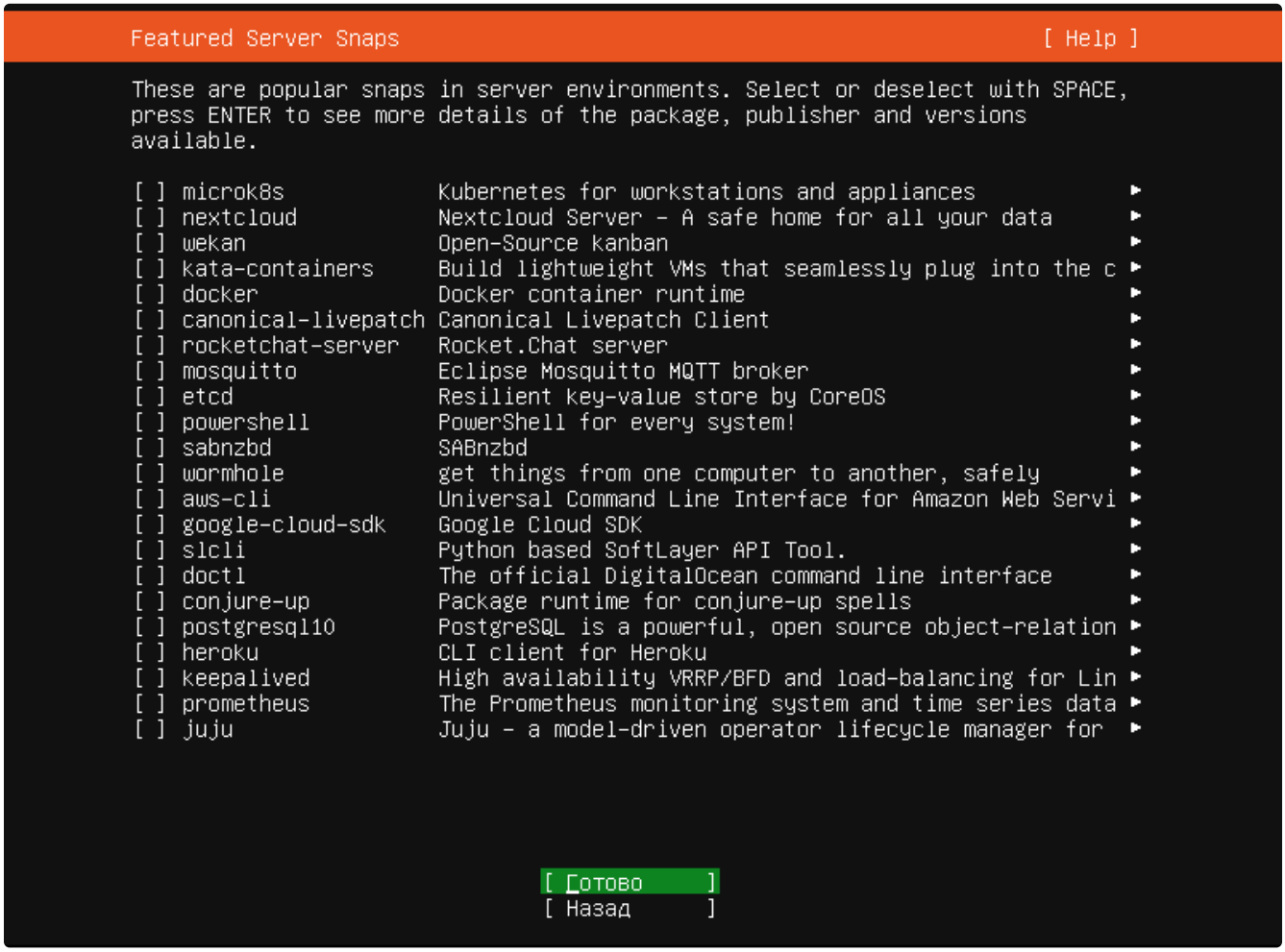
При установки выбираем Free установку:



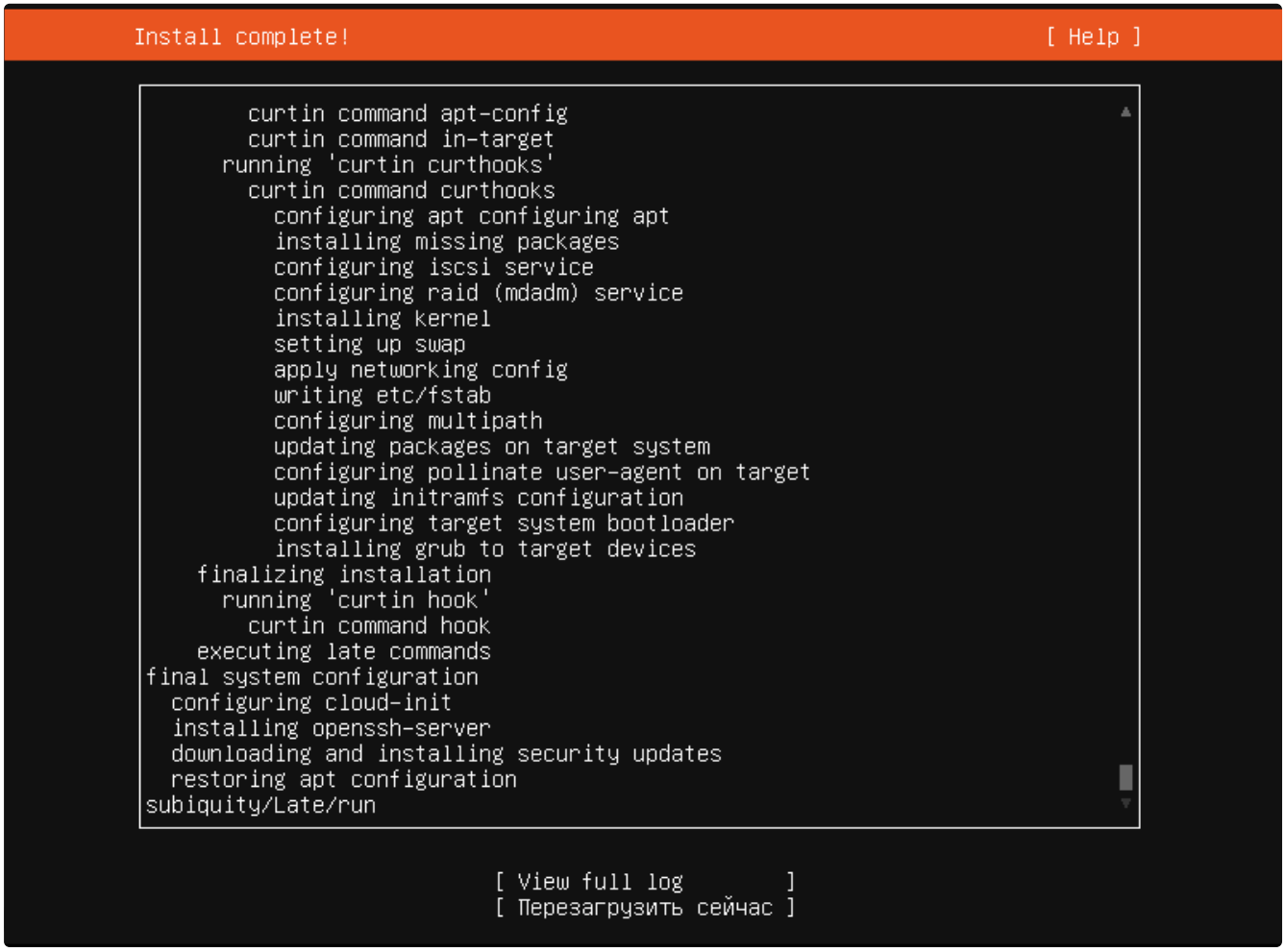
При инсталляции Ubuntu Linux, ставим openSSH, для удаленного управления



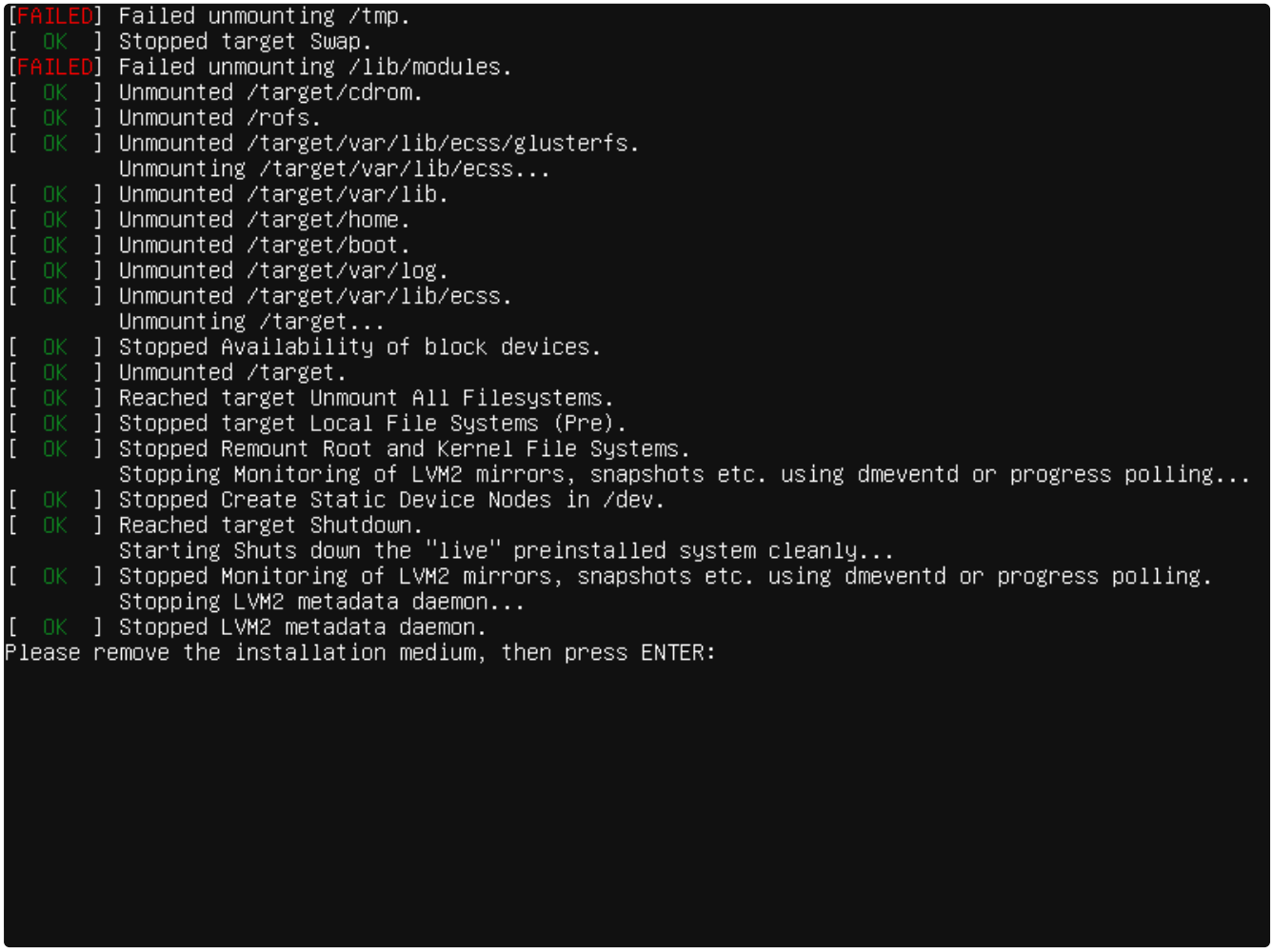
Подтвердить



Ждем появления сообщения - Перезагрузить сейчас



Извлекаем инсталляционный диск/флешку и Enter



Ubuntu Linux Server установлен

Таблица 1 — Вариант размещения информации в файловой системе на физических носителях для серверов

1	Загрузочный раздел операционной системы (создается автоматически)	boot	raid 1:hdd1,hdd2	boot	/boot	ext4	1 Gb	Первичный
2	Корневой раздел операционной системы	root	raid 1:hdd1,hdd2	root	/	ext4	30 Gb	Логический
3	Информация локальных баз данных	mnesia	raid 1:hdd1, hdd2	mnesia	/var/lib/ecss	ext4	10 Gb	Логический

4	Распределенная БД для хранения медиаресурсов	glusterfs	raid 1:hdd1, hdd2 или hdd3	glusterfs	/var/lib/ecss/restfs	ext4	Max Gb	Логический
5	Журналы функционирования подсистем ОС	log	raid 1:hdd1,hdd2 или hdd3	log	/var/log	ext4	20 Gb	Логический
6	Журналы функционирования подсистем ECSS	ecss_log	raid 1:hdd1,hdd2 или hdd3	ecss_log	/var/log/ecss	ext4	20 Gb	Логический
7	Базы данных	ecss_db	raid 1:hdd1,hdd2 или hdd3	ecss_db	/srv/ecss/ecss-postgres-bdr-ssw/	ext4	100–400 Gb*	Логический
8	Файлы пользователя	home	raid 1:hdd1,hdd2 или hdd3	home	/home	ext4	10 Gb	Логический

⚠ * Рекомендуемое значение для серий **Light, Light+, Midi** — 100 Gb. Рекомендуемое значение для серии **Heavy** — 200 Gb, **Super Heavy** — 400 Gb.

Для работы системы необходимо как минимум 256 Gb свободного пространства.

На серверах системы необходимо настроить параметр **"hostname".**

На всех серверах системы желательно указать одинаковое имя пользователя (любое, кроме **ssw**). Лицензия ECSS-10 привязывается к ключу eToken/ruToken и к имени компьютера (**hostname**). Системный пользователь **ssw** создается при инсталляции пакета **ecss-user**.

✔ Если используется один сервер, **рекомендуемое** значение hostname — **ecss1**;

9.5 Настройка сети

Согласно параметрам, заданным в техническом задании, выполним сетевые настройки. В данном примере считается, что необходимая операционная система уже установлена.

Рекомендуется разделять трафик, используемый в разных целях. Например, трафик управления и трафик VoIP. Для этого создаются 2 или более vlan. При небольшой нагрузке для простоты можно обойтись и одним vlan, но это будет в будущем доставлять неудобства при снятии дампов трафика и его анализе. В соответствии с техническим заданием на vlan настраиваются IP-адреса самого хоста, шлюзы, DNS и маршрутизация в другие сети.

В данном примере согласно ТЗ будем использовать адреса:

- 10.0.10.41/24 — для управления, vlan 10;
- 10.0.20.41/24 — для VoIP.

Внутри серверной платформы существует адресная структура и используются внутренние адреса для взаимодействия между подсистемами (нодами) в кластере. Например, внутренний адрес для кластера на одном сервере 127.0.0.1, при этом ядро (ecss-core) взаимодействует с сервером обработки мультимедиа-данных (ecss-media-server). Их взаимодействие происходит с использованием одного и того же адреса, но у каждой программной части свой транспортный порт: ecss-core — 5000, ecss-msr — 5040.

Для всех нод кластера определяется единый адрес для обращения к базе данных, например адрес 127.0.0.1. Таким образом, выполняется условие единообразия, при котором все ноды кластера имеют полностью одинаковые данные о текущем состоянии динамических составляющих программного коммутатора (например, история вызовов).

9.5.1 Подготовка сетевых интерфейсов системы

Согласно ТЗ, система имеет 4 сетевых интерфейса. Информацию об их состоянии можно посмотреть с помощью команды *ifconfig* или *ip a*:

```
eth0: flags=6211<UP,BROADCAST,RUNNING,SLAVE,MULTICAST> mtu 1500
ether 36:10:28:73:63:01 txqueuelen 1000 (Ethernet)

eth1: flags=6211<UP,BROADCAST,RUNNING,SLAVE,MULTICAST> mtu 1500
ether 36:10:28:73:63:01 txqueuelen 1000 (Ethernet)

eth2: flags=6211<UP,BROADCAST,RUNNING,SLAVE,MULTICAST> mtu 1500
ether be:77:ea:52:4d:39 txqueuelen 1000 (Ethernet)

eth3: flags=6211<UP,BROADCAST,RUNNING,SLAVE,MULTICAST> mtu 1500
ether be:77:ea:52:4d:39 txqueuelen 1000 (Ethernet)

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
```

Сначала настраиваются сетевые интерфейсы. В Ubuntu 22 для их настройки используется утилита netplan.

Данная утилита позволяет настроить сетевую конфигурацию и затем загрузить ее в систему с помощью сетевого менеджера networkd или NetworkManager.

```
sudo nano /etc/netplan/ecss_netplan.yaml
```

⚠ **Все другие файлы из этого каталога обязательно нужно переместить в другое место или удалить.**

В конфигурациях для каждого хоста в первую очередь мы объявляем секцию *ethernets,* в которой описываются существующие в системе ethernet-интерфейсы, используемые в дальнейшем. Для каждого интерфейса необходимо отключить использование динамического распределения адресов (DHCP).

Далее настраиваются vlan, на которых опционально определяются шлюзы для связи с внешним миром и адреса DNS-серверов, а также непосредственно IP-адреса для каждого интерфейса.

ВАЖНО

Обратите внимание, что во время редактирования netplan необходимо соблюдать правила разметки YAML:

- Обязательное наличие минимум двух пробелов перед каждой строкой (кроме network);
- Каждая подсекция дополнительно сдвигается на 2 пробела:

→ Секция |network
→ Подсекция |_'_'bonds:
→ Подсекция описания секции bonds |_'_'_'_'bonded_one:
→ и т.д. |_'_'_'_'...

- Перед знаком ":" нет пробела, после — один пробел;
- Перед знаком "-" количество пробелов такое, как будто начинается новый подраздел, после — один пробел.

Пример настройки файла ecss-netplan.yaml для варианта включения active-backup

```
# Netplan для хоста ecss1 программного коммутатора
# Обратите внимание на обязательное наличие минимум двух пробелов в каждой строке и
секции (кроме строки секции network)

network:
  version: 2 # Версия netplan
  renderer: networkd # Исполнитель конфигураций netplan
  ethernets: # Раздел описания интерфейсов ethernet
    enp0s3: # Название интерфейса ВМ к интернет
        dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
        dhcp6: no
        addresses: [192.168.56.41/24]
    enp0s8: # Название интерфейса для SSW
        dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
        dhcp6: no
  vlans:
    net.10: # Интерфейс управления
        id: 10
        link: enp0s3
        addresses: [10.0.10.41/24]
    net.20: # Интерфейс для VoIP
        id: 20
        link: enp0s8
        addresses: [10.0.20.41/24]
```

Следующие настройки **bonds** обязательны для сервера ECSS, чтобы схема работала верно:

mode: active-backup - задаёт режим работы, когда один из линков выбирается активным, а остальные остаются в резерве;
primary-reselect-policy: failure - указывает, что выбирать новый активный линк следует только тогда, когда текущий активный линк переходит в состоянии аварии. Это позволяет избежать лишних переключений;
gratuitous-arp: 5 - при смене активного линка в сторону коммутатора отправляются пять запросов gratuitous ARP, чтобы обновить на нём таблицу коммутации. Способствует более быстрому переключению;
all-slaves-active: true - заставляет принимать входящие кадры на backup интерфейсах. Таким образом балансировка трафика на MESe не мешает работе. Данные в сторону сервера идут со всех линков, а

сервер отправляет данные только с active линка;
mii-monitor-interval: 100 - активирует мониторинг линков через интерфейс MII и указывает интервал опроса в 100мс;
up-delay: 1000 - указывает считать поднявшийся интерфейс доступным для работы не сразу, а сделать задержку в одну секунду после того, как интерфейс поднялся. Необходимо для того, чтобы избежать лишних переключений в случае, когда порт "прыгает" несколько раз из состояния "включено" в состояние "выключено" и обратно.

Так же рекомендуется проверить отсутствие в каталоге **/etc/netplan/** еще каких либо файлов, если другие файлы присутствуют , то их нужно переместить в другой каталог или удалить, в противном случае возможна некорректная настройка сетевых интерфейсов и некорректная работа SSW.

Применим установленные параметры командой:

```
sudo netplan apply
```

На серверах системы необходимо настроить параметр **"hostname".**

На всех серверах системы желательно указать одинаковое имя пользователя (любое, кроме **ssw**). Лицензия ECSS-10 привязывается к ключу eToken/ruToken и к имени компьютера (**hostname**), поэтому необходимо использовать стандартные значения. Системный пользователь **ssw** создается при инсталляции пакета **ecss-user**.

Если используется один сервер, рекомендуемое значение hostname — **ecss1**;
Другие имена хостов возможны **только** при согласовании проекта, это потребуется для генерации лицензий.

Указать имя хоста: **ecss1** в файле /etc/hostname:

```
sudo nano /etc/hostname
```

Указать реальный Ipadd и имя хоста (для примера **10.0.20.41 ecss1**) в файле /etc/hosts:

```
127.0.0.1   localhost # Адрес локальной петли, используется некоторыми сервисами ecss
10.0.20.41  ecss1 # Адрес хоста
```

9.6 "Оптимизация" Операционной Системы

9.6.1 Выставить параметры ОС в режим производительности

Используем утилиту cpufrequtils.

```
sudo apt install cpufrequtils
```

по умолчанию после инсталляции Ubuntu использует режим "ondemand" - "по запросу" (производительность CPU по запросу приложений, экономит электроэнергию , но ниже производительность):

```
cat /etc/init.d/cpufrequtils | grep GOVERNOR=
```

в выходном сообщении системы , режим работы по умолчанию после инсталляции - "ondemand" :

```
GOVERNOR="ondemand"
```

Установить режим - результативность/производительность - в файле /etc/init.d/cpufrequtils значение "**ondemand**" заменить на "**performance**"

```
sudo sed -i 's/GOVERNOR="ondemand"/GOVERNOR="performance"/g' /etc/init.d/cpufrequtils
```

Перезапустить утилиту:

```
sudo /etc/init.d/cpufrequtils restart
```

Затем выполнить команду:

```
sudo systemctl daemon-reload
```

9.6.2 Отключить SWAP

Ubuntu сервер SSW работает **в реальном масштабе времени** ,
поэтому **все** необходимые **данные должны находится в оперативной памяти,** использование файла подкачки (swap-файл - /swap. img) может привести к увеличению времени обработки вызовов приложения SSW ECSS10 , **что недопустимо**. Swap - отключаем.
Выполнить последовательно три команды:

Отключение Swap:

```
sudo swapoff -a
```

Удалить файл swap.img.

```
sudo rm /swap.img
```

Закомментировать строку - /swap.img none swap sw 0 0 - в файле /etc/fstab -

привести ее к виду # /swap.img none swap sw 0 0

либо **удалить** эту строчку (/swap.img none swap sw 0 0)

```
sudo nano /etc/fstab
```

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
```

```
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/731728e2-4d6b-499a-afea-9362fd6726b2 / ext4 defaults 0 1
```

Для проверки выполните команду **free -h:**

```
free -h
```

Размер Swap равен 0 - то есть он выключен

```
free -h
               total        used        free      shared  buff/cache   available
Mem:           3,9G        110M        3,2G        820K        535M        3,5G
Swap:            0B          0B          0B
```

9.6.3 Установка часового пояса

При инсталляции Ubuntu-22 **не предлагается** установить часовой пояс(по умолчанию устанавливается время UTC). Его нужно установить **вручную** (для корректной работы системы тарификации , работ по расписанию и т.д.), например:

```
sudo timedatectl set-timezone Asia/Novosibirsk
```

9.6.4 Улучшение работы высоконагруженных серверов

Улучшить работу высоконагруженных серверов можно увеличив лимит открытых файлов.

Для установки лимита открытых файлов необходимо:

Проверить текущий лимит командой:

```
ulimit -a
```

результат :

```
eltex@ecss1:~$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
scheduling priority             (-e) 0
file size               (blocks, -f) unlimited
pending signals                 (-i) 15515
max locked memory       (kbytes, -l) 65536
max memory size         (kbytes, -m) unlimited
open files                      (-n) 1024
pipe size            (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
```

```
real-time priority              (-r) 0
stack size              (kbytes, -s) 8192
cpu time               (seconds, -t) unlimited
max user processes              (-u) 15515
virtual memory          (kbytes, -v) unlimited
file locks                      (-x) unlimited
```

Данного лимита (open files **1024**) **недостаточно** для нормальной работы высоконагруженных серверов.

Установить лимит открытых файл для каждого пользователя:

```
sudo sed -i  '55i\*                soft    nproc           65536\n*                hard
nproc           131072\n*               soft    nofile          65536\n*                hard
nofile          131072\nroot            -       memlock         unlimited' /etc/security/
limits.conf
```

Результат:

```
# /etc/security/limits.conf
#
#Each line describes a limit for a user in the form:
#
#<domain>        <type>  <item>  <value>
#
#Where:
#<domain> can be:
#        - a user name
#        - a group name, with @group syntax
#        - the wildcard *, for default entry
#        - the wildcard %, can be also used with %group syntax,
#                 for maxlogin limit
#        - NOTE: group and wildcard limits are not applied to root.
#          To apply a limit to the root user, <domain> must be
#          the literal username root.
#
#<type> can have the two values:
#        - "soft" for enforcing the soft limits
#        - "hard" for enforcing hard limits
#
#<item> can be one of the following:
#        - core - limits the core file size (KB)
#        - data - max data size (KB)
```

```
#        - fsize - maximum filesize (KB)
#        - memlock - max locked-in-memory address space (KB)
#        - nofile - max number of open files
#        - rss - max resident set size (KB)
#        - stack - max stack size (KB)
#        - cpu - max CPU time (MIN)
#        - nproc - max number of processes
#        - as - address space limit (KB)
#        - maxlogins - max number of logins for this user
#        - maxsyslogins - max number of logins on the system
#        - priority - the priority to run user process with
#        - locks - max number of file locks the user can hold
#        - sigpending - max number of pending signals
#        - msgqueue - max memory used by POSIX message queues (bytes)
#        - nice - max nice priority allowed to raise to values: [-20, 19]
#        - rtprio - max realtime priority
#        - chroot - change root to directory (Debian-specific)
#
#<domain>      <type>  <item>         <value>
#

#*               soft    core            0
#root            hard    core            100000
#*               hard    rss             10000
#@student        hard    nproc           20
#@faculty        soft    nproc           20
#@faculty        hard    nproc           50
#ftp             hard    nproc           0
#ftp             -       chroot          /ftp
#@student        -       maxlogins       4
*                soft    nproc           65536
*                hard    nproc           131072
*                soft    nofile          65536
*                hard    nofile          131072
root             -       memlock         unlimited

# End of file
```

ВАЖНО

Установку пакетов требуется делать НЕ из-под пользователя **ssw**.

9.7 Обновление программного обеспечения операционной системы

Для установки системы ECSS-10 необходимо добавить репозиторий ELTEX:

```
sudo sh -c "echo 'deb [arch=amd64] http://archive.eltex.org/ssw/jammy/3.18 stable main extras external' > /etc/apt/sources.list.d/eltex-ecss10-stable.list"
```

Далее необходимо выполнить импорт ключа командой:

```
sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 33CB2B750F8BB6A5
```

Перед началом установки необходимо обновить ОС:

```
sudo apt update
```

Если вы видите такое сообщение системы:

```
W: http://archive.eltex.org/ssw/jammy/3.18/dists/unstable/InRelease: Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.
```

Выполните следующую команду:

на обоих хостах

```
sudo cp /etc/apt/trusted.gpg /etc/apt/trusted.gpg.d
```

```
sudo apt upgrade
```

9.8 Установка и настройка программного обеспечения

установка пакетов deb

```
   ┌ версия ПО  ECSS.  Сейчас в заявке на сертификацию она заявлена как  3.

   │  Мы её менять не будем до следующей сертификации или
  │ выпуска принципиально новой версии системы.

   │    ┌ версия  System Release.  Общая версия для всех компонент входящих в конкретный релиз.

   │       │  Меняем её централизовано,
   │    │ принимаем решение о выпуске нового релиза.  Релизы между собой обычно не совместимы.
  ┌┴┐  ┌──┴──┐
ECSS.SysRel.SubMaj.SubMin
              └──┬──┘ └──┬──┘
        │     └ версия  Subsystem minor.  Минорную версию подсистемы устанавливает разработчик
подсистемы.

                  │           Минорную версию
        │      меняем при добавлении патча.  Как правило,  минорные версии в рамках одной мажорной
версии
        │      совместимы между собой и отличаются в рамках конкретных патчей.

                  └ версия  Subsystem major.  Мажорная версия устанавливается разработчиком
подсистемы.

                     Мажорную версию подсистемы необходимо менять при внесении в подсистему
существенных изменений.
```

Утилита установки пакетов APT анализирует версию пакета слева направо, для примера мы имеем пакет 14.14.7.7 ,
в репозитории находятся пакеты
14.14.7.8
14.14.7.9
14.14.8.1
14.14.20 -14.14.28
то при выполнении команды **sudo apt install имя пакета** → будет автоматически проверен и установлен пакет 14.14.28 , так как он самый последний, анализ будет выполнен по 3-й позиции (major), анализ по 4-й (minor) выполнен не будет (аналогичным образом будет выполнятся команда **sudo apt upgrade**).
В случае если для конкретной ситуации требуется перейти с версии 14.14.7.7 на версию 14.14.7.9 , стандартная команда → **sudo apt upgrade** , нам не поможет, так как будет выбран самый новый пакет, в данной ситуации нам **необходимо в явном виде** указать какую версию пакета мы хотим установить, в данном примере мы **должны** выбрать команду → **sudo apt install имя пакета=14.14.7.9** . Обычно это необходимо для тестирования определенного патча, для стандартных обновлений достаточно выбора привычной команды установки / обновления пакета

Устанавливаем все предложенные пакеты:

```
sudo apt install ntp ntpdate tcpdump vlan dnsmasq aptitude atop ethtool htop iotop mc minicom
mtr-tiny nmap pptpd pv screen ssh tftpd vim sngrep tshark cpanminus gnuplot libgraph-easy-perl
debconf-utils
```

Пакеты ПО которые рекомендуется установить для работы

Список **обязательного** сервисного программного обеспечения:

```
sudo apt install ntp tcpdump vlan dnsmasq
```

ntp	NTP-сервер
tcpdump	сниффер пакетов
vlan	управление VLAN
dnsmasq	легковесный DNS/DHCP-сервер

Список **рекомендуемого** диагностического и вспомогательного программного обеспечения:

```
sudo apt install aptitude atop ethtool htop mc screen ssh tftpd sngrep tshark gnuplot
libgraph-easy-perl debconf-utils iotop
```

aptitude	установка программ из репозиториев, рекомендуется использовать вместо программы apt/apt-get
atop	мониторинг загрузки хоста с функцией периодического сохранения информации в файлы
ethtool	просмотр статистики сетевых интерфейсов
htop	мониторинг процессов
mc	файловый менеджер
screen	мультиплексор терминалов
ssh	сервер и клиент SSH
tftpd	TFTP-сервер
sngrep	трассировка sip
tshark	консольный аналог wireshark
gnuplot	вывод графиков статистики
libgraph-easy-perl	Perl-модуль для преобразования или рендеринга графиков (в ASCII, HTML, SVG или через Graphviz)
debconf-utils	набор утилит для работы с базой debconf

iotop	инструмент для мониторинга использования ввода-вывода (IO) на диске в реальном времени в Linux

⚠ Данное программное обеспечение не требуется для работы системы ECSS-10, однако может упростить сервисное обслуживание системы и её отдельных компонентов со стороны инженеров эксплуатации и техподдержки.

⚠ Перед началом установки пакетов ecss , нужно убедится в соответствии полосы пропускания СПД необходимым требованиям.
Для этого выполнить команду **sudo ethtool <имя интерфейса>** для всех физических интерфейсов.
Пример:

```
sudo ethtool net.20
Settings for net.20:
    Supported ports: [ TP ]
    Supported link modes:   10baseT/Half 10baseT/Full
                            100baseT/Half 100baseT/Full
                            1000baseT/Full
    Supported pause frame use: No
    Supports auto-negotiation: Yes
    Supported FEC modes: Not reported
    Advertised link modes:  10baseT/Half 10baseT/Full
                            100baseT/Half 100baseT/Full
                            1000baseT/Full
    Advertised pause frame use: No
    Advertised auto-negotiation: Yes
    Advertised FEC modes: Not reported
    Speed: 1000Mb/s
    Duplex: Full
    Auto-negotiation: on
    Port: Twisted Pair
    PHYAD: 0
    Transceiver: internal
    MDI-X: off (auto)
    Link detected: yes
```

Проверить значение следующих параметров:
Advertised auto-negotiation: **Yes**
Speed: **1000Mb/s** (не менее)
Duplex: **Full**

9.8.1 Настройка пакета ecss-dns-env

Выполнить следующую команду:

```
sudo apt install -y ecss-dns-env
```

Настройщик предложит выбрать разделы для настройки, **не выбираем ничего** просто нажать **Enter**.

9.8.2 Установка Postgres

Примечание

Перед установкой `ecss-postgres-bdr-ssw` в системе, в `apt sources,` должен быть добавлен `docker` для этого выполнить следующие команды:

на обоих хостах

```
sudo install -m 0755 -d /etc/apt/keyrings
```

на всех хостах

```
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/
docker.asc
```

на всех хостах

```
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

на всех хостах

```
sudo sh -c "echo 'deb [arch=amd64 signed-by=/etc/apt/keyrings/docker.asc] https://
download.docker.com/linux/ubuntu   jammy stable' >> /etc/apt/sources.list.d/docker.list"
```

на всех хостах - Выполнить обновление:

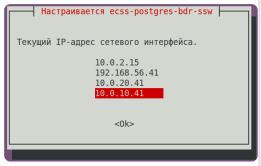
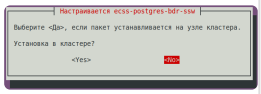
```
sudo apt update && sudo apt upgrade
```

9.8.3 Установка пакета ecss-postgres-bdr-ssw

ecss1

```
sudo apt install -y ecss-postgres-bdr-ssw
```

При конфигурации пакета **ecss-postgres-bdr-ssw** будут заданы вопросы:

Вопросы ecss-postgres-bdr-ssw	Ответы для ecss1	Пример
Текущий IP-адрес сетевого интерфейса.	**Выбрать** из списка IP-адрес для работы с Postgres	
Установка в кластере?	**No** (необходимо выбрать)	

Результат:

```
=======================================================
=======================================================
 ecss-postgres-bdr-ssw successfully installed/updated!
=======================================================
=======================================================
```

9.8.4 Установка ecss-node

Установка обязательного пакета **ecss-node** включает в себя установку и первоначальную настройку основных подсистем.

⚠ В системе уже должен быть установлен пакет **ecss-postgres-bdr-ssw**

Для установки пакета **ecss-node** выполните команду:

ecss1

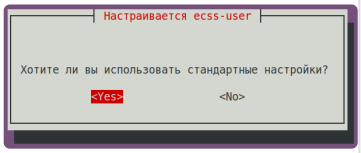
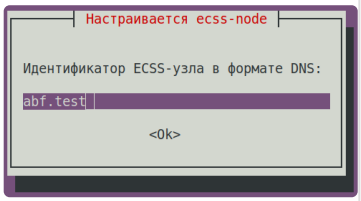
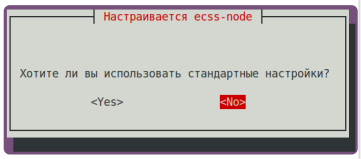
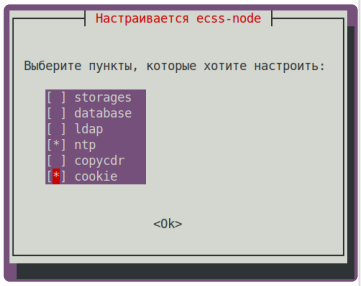
```
sudo apt install -y ecss-node
```

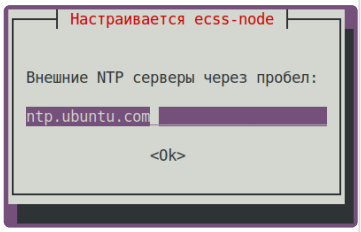
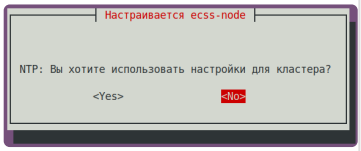
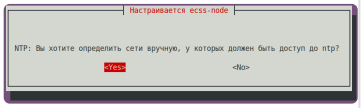
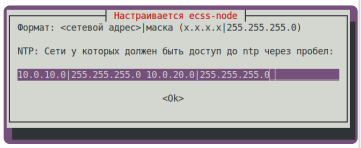
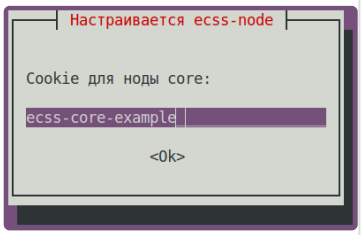
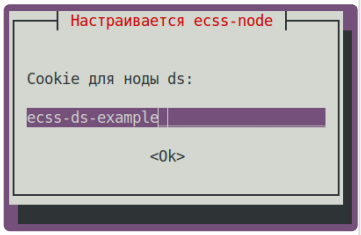
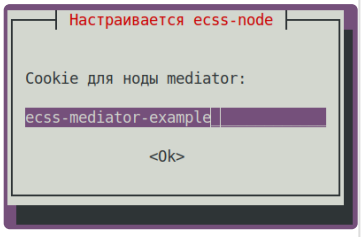
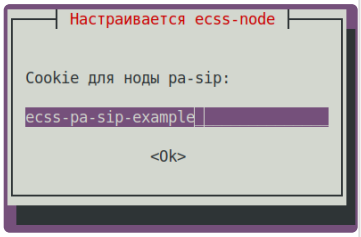
Во время установки пакета создается пользователь **ssw**, от имени которого запускаются все сервисы ecss*. Создаются необходимые каталоги, выполняется настройка DNS, идет настройка SSL-сертификатов.

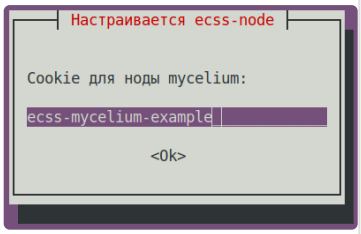
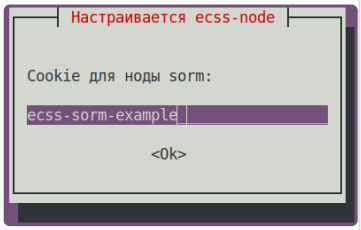
✓ При установки так же будет установлен пакет **ecss-user**

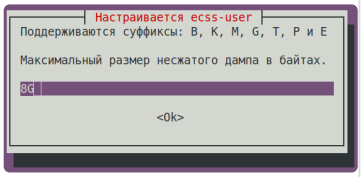
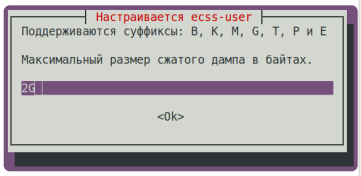
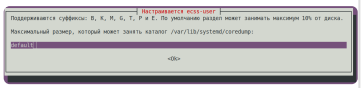
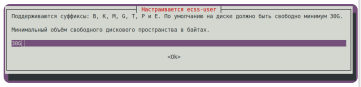
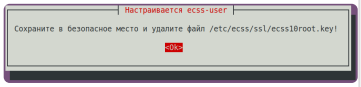
Установка на сервере ecss1

Во время установки будет предложено настроить параметры, необходимые для формирования конфигурационных файлов, примеры ответов следует ниже.

Вопросы ecss-user	Ответы для ecss1	Пример
Хотите ли вы использовать стандартные настройки?	Yes (значение по умолчанию)	
Вопросы ecss-node	**Ответы для ecss1**	**Пример**
Идентификатор ECSS-узла в формате DNS (Настройка имени кластера): (любое уникальное имя, в примере abf.test)	**abf.test** (необходимо ввести) Формат **ECSS_ID** Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире. `[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]`	
Хотите ли вы использовать стандартные настройки?	**No** (необходимо выбрать)	
Выберите пункты, которые хотите настроить:	**ntp + cookie** (необходимо выбрать)	

Внешние NTP серверы через пробел:	ntp.ubuntu.com (значение по умолчанию)	
NTP: Вы хотите использовать настройки для кластера?	**No** (необходимо выбрать)	
NTP: Вы хотите определить сети вручную, у которых должен быть доступ до ntp?	Yes (значение по умолчанию)	
NTP: Сети у которых должен быть доступ до ntp через пробел: (в примере 10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0)	**10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0** (необходимо ввести)	
Cookie для ноды core: (указать уникальный cookie для core , в примере ecss-core-example)	**ecss-core-example** (необходимо ввести)	
Cookie для ноды ds: (указать уникальный cookie для ds , в примере ecss-ds-example)	**ecss-ds-example** (необходимо ввести)	
Cookie для ноды mediator: (указать уникальный cookie для mediator , в примере ecss-mediator-example)	**ecss-mediator-example** (необходимо ввести)	
Cookie для ноды pa-sip: (указать уникальный cookie для pa-sip , в примере ecss-pa-sip-example)	**ecss-pa-sip-example** (необходимо ввести)	

Cookie для ноды mycelium: (указать уникальный cookie для mycelium , в примере ecss-mycelium-example)	**ecss-mycelium-example** (необходимо ввести)	
Cookie для ноды sorm: (указать уникальный cookie для sorm , в примере ecss-sorm-example)	**ecss-sorm-example** (необходимо ввести)	

Вопросы ecss-user	**Ответы для ecss1**	**Пример**
Максимальный размер несжатого дампа в байтах.	8G (значение по умолчанию)	
Максимальный размер сжатого дампа в байтах.	2G (значение по умолчанию)	
Максимальный размер, который может занять каталог /var/lib/systemd/coredump:	default (значение по умолчанию)	
Минимальный объем свободного дискового пространства в байтах.	30G (значение по умолчанию)	
Сохраните в безопасное место и удалите файл /etc/ecss/ssl/ecss10root.key!	Ok (значение по умолчанию)	

Проверить статус сервисов следующей командой:

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
```

В случае статуса "active" продолжаем дальше, в случае статуса "inactive" выполнить команду "sudo systemctl restart ecss-<имя пакета>"

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
active
active
active
```

```
active
active
```

Для просмотра информации о состоянии синхронизации с NTP, используется команда ntpq −p . Если использовать дополнительный ключ −n, вместо имени сервера будет указан IP-адрес:

```
ntpq -p
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
*185.125.190.56  17.253.28.253    2 u   45   64    1   83.893   +0.470   2.177
```

9.8.5 Установка лицензии

Система ECSS-10 использует ELM-лицензирование – сервис распределения лицензий ECSS-экосистемы, состоит из **ecss-license-agent** и **ecss-license-master** .

- **ECSS License Agent** - Адаптер для работы с EcssLicenseMaster (LM). Получает список хостов EcssLicenseMaster, реализует логику подключения и переподключения к LM.
  - Отвечает за получение, валидацию и дальнейшую отправку лицензий клиенту.
  - Отправляет статусы подключения и полученные лицензии клиенту.
  - Предоставляет API для отправки HTTP запросов и WS событий на LM.
- **Ecss License Master** - Сервис оркестрации параметров лицензии для ECSS-узлов.
  - Отвечает за:
    - Получение лицензии для конкретного ECSS-узла (SSW / AuP) либо устройства (SMG)
    - Распределение параметров лицензии между ECSS-узлами и ECSS-сервисами
    - Контроль активности ECSS-узлов и их валидация во избежание тиражирования лицензии сверх указанных лимитов
    - Контроль изменения указанных лимитов на ECSS-узлах, и их динамическое распределение в соответствии с текущей нагрузкой.

ecss-license-agent является составной частью SSW, входит в пакет ecss-node. Отдельной конфигурации не требует.

В свою очередь **Ecss License Master** может быть установлен как на SSW так и на отдельно выделенный сервер. В случае если Ecss License Master уже установлен на вашем проекте, то потребуется только дополнить запись о новом компоненте в конфигурационном файле LM, и следующий пункт – установка "Установка LM" не потребуется.

Установка LM на SSW host

Для установки пакета **ecss-license-master** необходимо добавить репозиторий следующей командой:

```
sudo sh -c "echo deb [arch=amd64] https://archive.eltex.org/ssw/jammy/aup/2.0 stable main
extras external >> /etc/apt/sources.list.d/lm-stable.list"
```

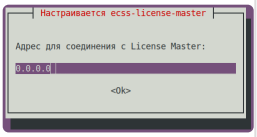
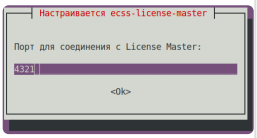
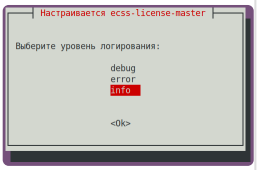
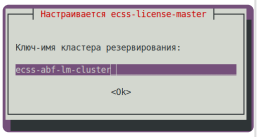
Выполнить обновление из добавленного репозитория командой:

```
sudo apt update
```

Установить пакет LM (ecss-license-master)

```
sudo apt install ecss-license-master
```

При установке пакета ecss-license-master будут заданы вопросы:

Вопросы	Ответы	Примеры
Адрес для соединения с License Master: License Master listen ip:	0.0.0.0 (значение по умолчанию)	
Порт для соединения с License Master: License Master listen port:	4321 (значение по умолчанию)	
Выберите уровень логирования: Log level:	info (значение по умолчанию), возможные варианты: debug, error, info	
Ключ-имя кластера резервирования: License Master cluster key:	указать уникальное имя ecss-lm-cluster (значение по умолчанию) Менеджеры с одинаковыми именами попытаются собраться в кластер и резервировать друг друга.	

Данные сохраняются в файле - **/etc/ecss/ecss-license-master/config.env** :

```
ECSS_LM_HOST=0.0.0.0
ECSS_LM_PORT=4321
ECSS_LM_LOG_LEVEL=info
RELEASE_COOKIE=ecss-abf-lm-cluster
```

Конфигурация LM

Сервис ecss-license-master использует два конфигурационных файла: **/etc/ecss/ecss-license-master/config.env** и **/etc/ecss/ecss-license-master/config.yaml**.
В файл **config.env** информация сохраняется в момент установки пакета **ecss-license-master**.

Config.yaml файл содержит основные настройки для подключения к ELM серверу, откуда брать лицензии, какие лицензии, и куда их потом передавать, **все это нужно сконфигурировать вручную**.

При установке LM на SSW

Выполнить команду:

```
sudo nano /etc/ecss/ecss-license-master/config.yaml
```

информация в файле после установки

```
elm_addresses: []
licenses: []
ecss_nodes: []
```

В поле **elm_adresses** нужно прописать список серверов ELM к которым мы хотим подключаться для получения лицензий. Первый адрес в списке используется как основной. Дополнительные строки с адресами работают как резерв.

Данные ELM сервера(ов) получить у менеджера проекта.

пример:
- "elm-1.eltex.loc:8099"
- "192.168.111.22:8099"
- "elm-3.eltex.loc:8099"

Дальше задаются сами лицензии в поле **licenses**.

- **id** – может быть любой, главное чтоб был уникальный у каждой лицензии. Влияет только на адрес лицензии на самом LM (например 0,1,2,3).
- **kind** – для SSW поддерживается только ***ssw.***
- **type** – выбирается как ***elm*** если запрашивать данную лицензию нужно с ELM сервера. Если нужно использовать файловую лицензию SSW, то тип выбирается как ***ecss_license***.
  - при ***elm*** типе, необходимо задать поля **license_key** и **product_id** для идентификации лицензии.
  - при ***ecss_license*** типе, задается поле **license** содержащее ключ статичной лицензии

Параметры **license_key** и **product_id** должны быть согласованы с данными загруженными на ELM сервере .

Значение параметров **license_key**, **product_id** или **license**, **passport** нужно получить у менеджера проекта.

Блок **ecss_nodes**.
В нем нужно задать данные сервисов которые будут получать лицензию. В нашем случае SSW.
Сервис задается тремя параметрами:

- **id –** идентификатор сервиса, должен совпадать с ECSS_ID лицензируемого SSW.
- **kind –** тип лицензируемого сервиса. по аналогии с блоком лицензий, для SSW поддерживается только ***ssw.***

- **license_id –** идентификатор лицензии которая будет отправляться на данный сервис. Соответствует идентификаторам лицензии из блока **licenses**

⚠ Два SSW с **разными ECSS_ID** не могут получить одну и ту же лицензию, конфигурация с таким распределением будет помечена некорректной.
Кроме того, два SSW с **одинаковыми ECSS_ID** не смогут одновременно получать одну и ту же лицензию с одного LM,
поэтому крайне важно корректно устанавливать свой ECSS_ID и делать его уникальным как минимум в зоне видимости используемого LM.

✓ Формат **ECSS_ID**
Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире.

```
[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]
```

⚠ Для SSW работающего в кластере указываем только одну запись в блок **ecss_nodes** на кластер, индивидуально для каждого хоста указывать нельзя. Иначе будут генерироваться два запроса с одинаковым ECSS_ID что недопустимо.

Пример: файла config.yaml

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
   kind: SSW
   type: elm
   license_key: "ssw1test"
   product_id: "ECSS1000001"
 - id: 1
   kind: ssw
   type: ecss_license
   license: "af615ebb92d381125ff"
ecss_nodes:
 - id: abf.test
   kind: ssw
   license_id: 0
```

Выполнить перезагрузку сервиса **ecss-license-master.service** для обновления конфигурации из файла **/etc/ecss/ecss-license-master/config.yaml** следующей командой:

если кластер, то на обоих хостах

```
sudo systemctl restart ecss-license-master.service
```

При подключении SSW к ранее установленному LM

В случае если LM уже установлен и используется в локальной сети требуется добавить параметры нового SSW в конфигурационный файл **/etc/ecss/ecss-license-master/config.yaml** в блоках licenses и ecss_nodes.

Выполнить команду:

если кластер, то на обоих хостах

```
sudo nano /etc/ecss/ecss-license-master/config.yaml
```

для примера:

Было:

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
   kind: SSW
   type: elm
   license_key: "ssw1test"
   product_id: "ECSS1000001"
 - id: 1
   kind: ssw
   type: ecss_license
   license: "af615ebb92d381125ff"
 - id: 2
   kind: SSW
   type: elm
   license_key: "ssw2test"
   product_id: "ECSS1000002"
ecss_nodes:
 - id: 1.test
   kind: ssw
   license_id: 0
 - id: 2.test
   kind: ssw
   license_id: 1
 - id: 3.test
   kind: ssw
   license_id: 2
```

Стало:

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
```

```
    kind: SSW
    type: elm
    license_key: "ssw1test"
    product_id: "ECSS1000001"
  - id: 1
    kind: ssw
    type: ecss_license
    license: "af615ebb92d381125ff"
  - id: 2
    kind: SSW
    type: elm
    license_key: "ssw2test"
    product_id: "ECSS1000002"
  - id: 3
    kind: SSW
    type: elm
    license_key: "ssw3test"
    product_id: "ECSS1000003"
ecss_nodes:
  - id: 1.test
    kind: ssw
    license_id: 0
  - id: 2.test
    kind: ssw
    license_id: 1
  - id: 3.test
    kind: ssw
    license_id: 2
  - id: abf.test
    kind: ssw
    license_id: 3
```

Выполнить перезагрузку сервиса **ecss-license-master.service** для обновления конфигурации из файла **/etc/ecss/ecss-license-master/config.yaml** следующей командой:

```
sudo systemctl restart ecss-license-master.service
```

Подключение SSW к Менеджера Лицензий

Выполнить конфигурацию подключения SSW к LM следующей командой в CoCon, (в примере LM , IPadd 10.0.10.51. значение порта из файла /etc/ecss/ecss-license-master/config.env по умолчанию 4321):

выполняется в CoCon

```
/system/licence/manager/set --hosts [https://10.0.10.51:4321]
```

В случае установки LM на SSW указывать IP адреса OAM

После выполнения проверить статус подключения командой:

выполняется в CoCon

```
/system/licence/manager/show-status
┌──────────────────────────┬───────┬─────┐
│          Host            │Current│Alive│
├──────────────────────────┼───────┼─────┤
│https://10.0.10.51:4321   │*      │true │
└──────────────────────────┴───────┴─────┘
```

Должны иметь статус alive=true, и быть "current".

- Current - отображает к какому хосту подключён SSW. Если есть подключение, указывается *, если нет - ничего не указано.
- Alive - отображает статус доступности хоста (healthcheck).

Подключение к License Manager

Отсутствие подключения к License Manager \ ELM будет равноценно отсутствию доступа к токену, что приведёт к критическим авариям, переходу в аварийный режим по истечению которого, если связь не будет восстановлена, система перейдёт на дефолтную лицензию.

данное поведение применимо только при работе в **elm** режиме

Если статус к LM показан current, можно посылать запрос на загрузку лицензии на SSW командой:

выполняется в CoCon

```
/cluster/storage/ds1/licence/request
```

пример выполнения команды /cluster/storage/ds1/licence/request

type=elm

```
/cluster/storage/ds1/licence/request
Licence received
[*******                                                    ] 6s 2ms
Success: Licence parameters applied
```

В режиме "type=ecss_license" команда **/cluster/storage/ds1/licence/request** выполнится только в случае если был предварительно загружен паспорт.

type=ecss_license

```
/cluster/storage/ds1/licence/request
Waiting for licence...
[*******                                                    ] 6s
┌─┬──────────────────────────────────────────────────────┬─────────┬──────────────────────────────
──┐
│A│                 Description                          │Old Value│              New Value
  │
```

```
|-|-----------------------------------------------------------|---------|-------------------
-----------|
|^|Support for reservation of call-processes                  |false    |true
           |
|^|Maximum call duration (in seconds)                         |60       |2678400
           |
|^|Maximum number of simultaneous calls                       |5        |10000
           |
|^|Elph                                                       |         |
           |
|^| total count of members of the one Elph group              |10       |100
           |
|^|Subscribers limit                                          |10       |500000
           |
|^|Virtual subscribers limit                                  |infinity |1000
           |
|^|Add-on conferences                                         |         |
           |
|^| total count                                               |30       |1000
           |
|^|Add-on conferences                                         |         |
           |
|^| members of the one add-on conference                      |16       |300
           |
|^|Chat rooms                                                 |         |
           |
|^| total count                                               |30       |100
           |
|^|Chat rooms                                                 |         |
           |
|^| members of the one chatroom                               |16       |300
           |
|^|Meet Me                                                    |         |
           |
|^| total count                                               |0        |100
           |
|^|Meet Me                                                    |         |
           |
|^| members of the one Meet Me conference                     |0        |300
           |
|^|SORM                                                       |         |
           |
|^| enabled                                                   |false    |true
           |
|+| channels on SORM mediator                                 |         |64
           |
|^|Sorm extractor                                             |none     |mfi
           |
|^|Support antifraud system                                   |none     |custom
           |
|+|Call center                                                |         |
           |
|+| operator's capability for look at calls in queue          |         |true
           |
|+| operator's capability for use Intervension SS             |         |true
           |
|^|Call center                                                |         |
           |
```

```
|^| active agents                                          |0        |1000
              |
|^|Call center                                            |         |
              |
|^| active supervisors                                     |0        |100
              |
|^|Teleconference                                          |         |
              |
|^| channels                                               |3        |2000
              |
|^|Teleconference                                          |         |
              |
|^| max members of one teleconference                      |0        |200
              |
|^|Teleconference                                          |         |
              |
|^| active count                                           |0        |32
              |
|^|TSMN system                                             |         |
              |
|^| concurrent calls (active)                              |0        |50
              |
|^|TSMN system                                             |         |
              |
|^| concurrent calls (backup)                              |0        |50
              |
|^|TSMN system                                             |         |
              |
|^| backup mode                                            |none     |"backup"
              |
|^|Total count of simultaneous records voice calls         |0        |200
              |
|+|IVR                                                     |         |
              |
|+| customization enabled                                  |         |true
              |
|+| Automatic Speech Recognition (ASR) subsystem enabled|         |true
              |
|^|Channels on dialer outgoing calls                       |2        |5
              |
|+|Supplementary Services licence package(s)               |         |
              |
|+| name                                                   |         |'ECSS-ADV'
              |
|+| limit                                                  |         |10000
              |
|+| description                                            |         |"Additional services of 4
and 5 levels" |
|+| SS list                                                |         |[1,2,3,4,5]
              |
|+| name                                                   |         |'ECSS-BAS'
              |
|+| limit                                                  |         |10000
              |
|+| description                                            |         |"Basic services of 1 and
2 levels"      |
|+| SS list                                                |         |[1,2]
              |
```

```
|+| name                                                    |         |'ECSS-BAS+'
              |
|+| limit                                                   |         |10000
              |
|+| description                                             |         |"Expansion of basic
services of 3 level"|
|+| SS list                                                 |         |[1,2,3]
              |
|^|GEO backup                                              |         |
              |
|^| enabled                                                 |false    |true
              |
|^|SIGTRAN                                                  |         |
              |
|^| SEP                                                     |false    |true
              |
|^| STP                                                     |false    |true
              |
└─┴─────────────────────────────────────────────────────────┴─────────┴──────────────────────
──────────────┘
Legend:
    '^' - Changed;
    '-' - Deleted;
    '+' - New.


[request] You are trying to add a license that is different from the current one by the pa
          rameters in the table (other values will remain unchanged).
Licence received      o ?> yes
[*******                                                              ] 1m 15s
Success: Licence parameters applied
```

В зависимости от выбранного типа лицензии для SSW : "type=**elm**" или "type=**ecss_license**" на команду просмотра лицензий получим разный вывод:

type=ecss_license

```
/cluster/storage/ds1/licence/list-licence
┌─┬────────────────────┬──────────┬──────┬──────────────────┬──────────────────────┬──────
──────────────────┬──────────┐
|Id|Creation date(UTC) |   SSW ID   |Active|  Description   |Comm. Exp.
Date(UTC)|Expiration date(UTC)|Time left|
├─┼────────────────────┼──────────┼──────┼──────────────────┼──────────────────────┼──────
──────────────────┼──────────┤
|1 |12.03.2026 18:40:00|ECSS 010070 |*     |ECSS TPM License|13.03.2027 06:59:59
|14.03.2026 10:44:05 |1d 6h 59m|
|0 |01.01.1990 00:00:00|ECSS DEFAULT|      |Default licence |                      |
|          |
└─┴────────────────────┴──────────┴──────┴──────────────────┴──────────────────────┴──────
──────────────────┴──────────┘
```

TPM - является полным аналогом лицензирования которое использовалось раньше, с токеном и паспортом, без подключения к LM.

или

type=elm

```
/cluster/storage/ds1/licence/list-licence
┌──┬────────────────────┬──────────┬──────┬──────────────┬────────────────────┬──────────┐
│Id│Creation date(UTC) │   SSW ID   │Active│  Description   │Comm. Exp. Date(UTC)│Expiration date(UTC)│Time left│
├──┼────────────────────┼──────────┼──────┼──────────────┼────────────────────┼──────────┤
│1 │03.02.2026 16:26:34│ECSS2000009 │*     │ECSS ELM License│03.02.2027 16:26:34 │14.03.2026 10:49:12 │1d 6h 59m│
│0 │01.01.1990 00:00:00│ECSS DEFAULT│      │Default licence │                    │          │
└──┴────────────────────┴──────────┴──────┴──────────────┴────────────────────┴──────────┘
```

9.8.6 Установка пакетов ecss-restf, ecss-media-server, ecss-web-conf

Выполнить конфигурацию для Sip адаптера , выполнив следующую команду:

```
sudo systemctl edit ecss-pa-sip.service
```

Добавить в конфигурационный файл следующие данные:

```
[Service]
LimitNOFILE=65536
```

Вставить эти две строчки необходимо в определенное место, (начиная с Ubuntu 22.04.2 есть несколько важных моментов. Файл, который открывается, полностью закомментирован, но первые несколько строк выглядят так (см рисунок ниже). Любые вносимые вами правки должны быть между этими двумя блоками комментариев.)

```
 GNU nano 6.2                                    /etc/systemd/system/ecss-pa-sip.service.d/.#override.confa280cb6585b7c
### Editing /etc/systemd/system/ecss-pa-sip.service.d/override.conf
### Anything between here and the comment below will become the new contents of the file

[Service]
LimitNOFILE=65536

### Lines below this comment will be discarded

### /lib/systemd/system/ecss-pa-sip.service
# [Unit]
# Description=daemon ecss-pa-sip-14.16.0.346 of ecss-10
# After=ntp.service
# After=ecss-mycelium.service ecss-ds.service ecss-mediator.service ecss-core.service dnsmasq.service epmd.service ecss.slice
# Requires=epmd.service ecss.slice
# StartLimitIntervalSec=120
# StartLimitBurst=4
#
# [Service]
# Slice=ecss-erlang.slice
# ExecStart=/usr/sbin/ecss-pa-sip %i
```

Перезагрузите конфигурацию

```
sudo systemctl daemon-reload
```

Актуализировать конфигурационный файл, выполнив перезагрузку сервера:

```
sudo systemctl restart ecss-mycelium ecss-ds ecss-core ecss-pa-sip ecss-mediator
```

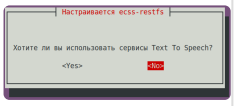
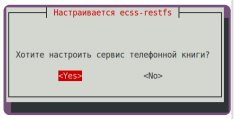
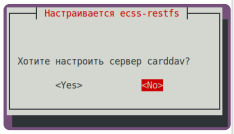
Далее устанавливаются пакеты *ecss-restf затем ecss-media-server, ecss-media-resources, , ecss-web-conf* и другие в любом порядке:

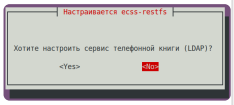
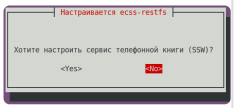
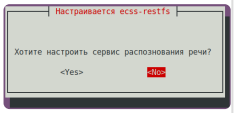
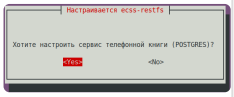
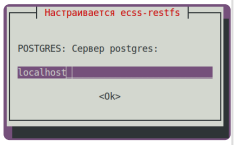
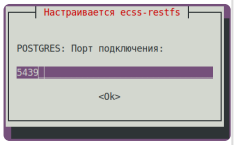
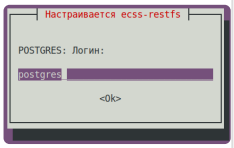
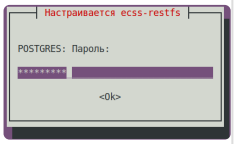
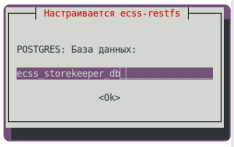
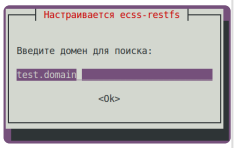
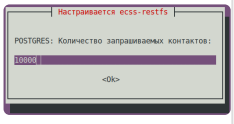
ecss-restfs

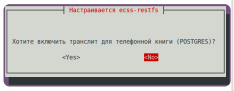
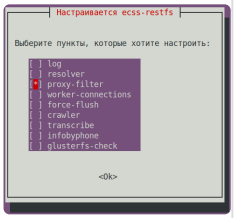
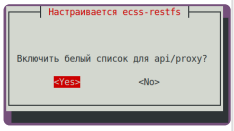
```
sudo apt install -y ecss-restfs
```

Установка **ecss-restfs.**

При его установке будет предложено настроить конфигурацию при помощи вопросов: , также инсталлятор предложит установить и настроить пакет Text2speech от Yandex.

Вопросы ecss-restfs	**Ответы**	**Пример**
Хотите ли вы использование функции Text To Speeh (Use TTS service)	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (Configure phone book)	**Yes** (нужно указать)	
Хотите настроить сервер carddav?	No (значение по умолчанию)	

Вопросы ecss-restfs	Ответы	Пример
Хотите настроить сервис телефонной книги (LDAP)?	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (SSW)?	No (значение по умолчанию)	
Хотите настроить сервис распознавания речи (Configure speech recognition service)	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (POSTGRES)?	Yes (значение по умолчанию)	
POSTGRES: Сервер postgres:	localhost (значение по умолчанию)	
POSTGRES: Порт подключения:	5439 (значение по умолчанию)	
POSTGRES: Логин:	postgres (значение по умолчанию)	
POSTGRES: Пароль:	postgres1 (значение по умолчанию)	
POSTGRES: База данных:	ecss_storekeeper_db (значение по умолчанию)	
Введите домен для поиска:	test.domain (значение по умолчанию)	
POSTGRES: Количество запрашиваемых контактов:	10000 (значение по умолчанию)	

Вопросы ecss-restfs	Ответы	Пример
Хотите включить транслит для телефонной книги (LDAP)?	No (значение по умолчанию)	
Выберите пункты, которые хотите настроить:	**proxy-filter** (необходимо выбрать)	
Включить белый список для api/proxy?		

белый список адресов прописать в файл /etc/ecss/ecss-restfs/api-proxy.conf

```
# ------------------------------------------------------------
# Формат файла фильтра API (api/proxy):
#   - IP: 192.168.0.1
#   - Домен: example.com
#   - Wildcard: *.wildcard.com
#   - Комментарии: строки, начинающиеся с #
# Пример:
#   192.168.0.1
#   example.com
#   # комментарий
#   *.wildcard.com
# ------------------------------------------------------------
```

ВАЖНО!

В случае успешного проксирование (200) кеш хранится 28 дней, поэтому для применения конфигурации фильтра, в случае изменения, - кеш (/var/cache/ecss/restfs-api) стоит удалить и перезапустить сервис ecss-restfs.

после установки пакета ecss-restfs проверьте наличие wav файлов автоинформатора в директории /var/lib/ecss/restfs/system/sounds/ командой :

```
ll /var/lib/ecss/restfs/system/sounds/
```

наличие порядка 140 wav файлов в указной выше директории говорит о корректности установки пакета ecss-restfs

Отсутствия wav файлов в указной выше директории, говорит о проблеме при установки пакета ecss-restfs. Для ее решения выполните команду:

```
sudo apt --fix-broken install
```

и повторите команду проверки:

```
ll /var/lib/ecss/restfs/system/sounds/
```

осталось проверить доступность этих файлов извне, для этой проверки выполним команду:

```
wget http://ecss1:9990/system/sounds/ai_you.wav
```

```
wget http://ecss1:9990/system/sounds/ai_you.wav
--2023-12-18 17:43:29--  http://ecss1:9990/system/sounds/ai_you.wav
Resolving ecss1 (ecss1)... 127.0.1.1, 10.0.10.11
Connecting to ecss1 (ecss1)|127.0.1.1|:9990... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11670 (11K) [audio/x-wav]
Saving to: ‘ai_you.wav’

ai_you.wav                                          100%
[=======================================================================================>]
11,40K  --.-KB/s    in 0s

2023-12-18 17:43:29 (301 MB/s) - ‘ai_you.wav’ saved [11670/11670]
```

проверочный файл можно удалить

```
rm ai_you.wav
```

ecss-media-server

```
sudo apt install -y ecss-media-server
```

Для медиасервера (*ecss-media-server/MSR*) возможно начальное конфигурирование с записью параметров в файл конфигурации,
для этого нужно провести конфигурацию **transport bind-addr,mcc bind-addres**:

Вопросы ecss-media-server	**Ответы для ecss1**
[MSR SIP] Введите bind-ip адрес (Enter)	**10.0.20.41** (необходимо ввести)
[MSR Control-Channel] Введите bind-ip-адрес	**10.0.20.41** (необходимо ввести)

После формирования конфигураций по умолчанию производим проверку:

```
cat /etc/ecss/ecss-media-server/config.xml
```

Внутри лежит конфигурация для msr: config.xml, в директории conf.d лежит конфигурация default.xml.

По своей сути default.xml — это дополнение config.xml, которое определяет секцию аккаунтов. Это сделано для того, чтобы после обновлений пакета данная конфигурация оставалась неизменной. Вид config.xml представлен здесь: файл конфигурации.

```
<?xml version="1.0" encoding="utf-8"?>
<config date="09:24:23 03.12.2025">
  <general log-level="3" log-rotate="yes" max-calls="2148" max-vid-calls="100" max-in-group="51
2" load-sensor="media" load-delta="10" calls-delta="100" spool-dir-size="100M" log-name="msr.lo
g" log-path="/var/log/ecss/media-server" use-srtp="disabled" enable-ice-transport="no" ice-
update="no" aggressive-ice="yes" stun-server="" suspicious-mode="no"/>
  <transport bind-addr="10.0.20.41" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca" user="webrtc@live.com" password="muazkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60" rtcp-timeout="0" rtp-
timeout="350" udp-src-check="no" cn-multiplier="3" port-start="12000" port-range="4496" tias-
in-sdp="no" thread-cnt="2" vid-enc-threads="2" vid-dec-threads="2" video-conf-layout="evenly"
 keyframe-interval="1000" vid-decode-delay="100" silent-codec-switch="yes" silence-threshold="-
30" dtmf-flash-disable="no" video-dscp="0" other-dscp="0" dummy-video-src="/usr/share/ecss-
media-server/video/dummy_video.yuv" video-enc-width="640" video-enc-height="360" finalsilence="
1000" rtcp-stat-dump="yes" dtmf-tg-fpc-loop="10" dtmf-tg-fit="1" dtmf-tg-fot="2" dtmf-tg-
volume="12288"/>
  <codec pcma="1" pcmu="2" ilbc="0" gsm="0" g722="3" g726="0" g729="0" speex="0" l16="0" g7221="
0" opus="0" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
  <accounts>
    <!-- <dynamic msr_name="msr.name"
            realm="sip:127.0.0.1:5000"
            dtmf_mode="rfc+inband+info"
            auth_name="user"
            auth_password="password" /> -->
  </accounts>
  <pbyte>
    <mcc bind-addr="10.0.20.41" port="5700"/>
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <metrics enable="yes" use-pushgateway="yes" pushgateway-addr="127.0.0.1:9091" push-interval="
5" port="8100" msr_name="msr2"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

по умолчанию , после инсталляции ECSS-10 **активизирует только** кодеки - **pcma, pcmu, g722,** h264, h263-1998, t38. Остальные имеют статус =0 = выкл. Если хотите активизировать остальные кодеки, в редакторе nano измените приоритет с 0 на цифру приоритета (уровень приоритета не должен повторятся для сервиса audio/video/fax)
sudo nano /etc/ecss/ecss-media-server/config.xml

чтобы активизировать **все** кодеки
строчку -
<codec pcma="1" pcmu="2" ilbc="**0**" gsm="**0**" g722="3" g726="**0**" g729="**0**" speex="**0**" l16="**0**" g7221="**0**" opus="**0**" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
привести к виду -
<codec pcma="1" pcmu="2" ilbc="**6**" gsm="**4**" g722="3" g726="**11**" g729="**5**" speex="**7**" l16="**8**" g7221="**9**" opus="**10**" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>

Недопустимо указывать у разных кодеков одинаковый приоритет, для примера g722="5" g729="5". В этом случае они будут работать не корректно/игнорироваться.
За исключением "0" - выключено.

Рассмотрим секцию аккаунтов (файл default.xml):

```
cat /etc/ecss/ecss-media-server/conf.d/default.xml
```

Проверить на корректность данных.

Настройка msr для ecss1(/etc/ecss/ecss-media-server/conf.d/default.xml)

```
<?xml version="1.0"?>
<config>
    <accounts>
        <dynamic msr_name="msr.ecss1" realm="sip:10.0.20.41:5000" dtmf_mode="rfc+inband+info"
 auth_name="user" auth_password="password">
            <via dynamic-ifaces="no">
                <iface name="net.20" ip="10.0.20.41"/>
            </via>
        </dynamic>
    </accounts>
</config>
```

В ней указаны текущие настройки, согласно которым происходит регистрация msr на core.

Основными параметрами здесь являются: msr_name и realm:

- msr_name — параметр, определяющий название msr. (рекомендуется задавать название msr. и к какому хосту он принадлежит, например msr.ecss1);
- realm — определяет адрес для регистрации на ядре. Точка входа по умолчанию: порт 5000, адрес: 127.0.0.1.

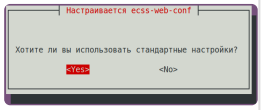
После изменения конфигурации файлов **/etc/ecss/ecss-media-server/config.xml** и **/etc/ecss/ecss-media-server/conf.d/default.xml** необходимо выполнить перезапуск сервиса **ecss-media-server**, чтобы изменения вступили в силу, следующей командой:

```
sudo systemctl restart ecss-media-server
```

ecss-web-conf

```
sudo apt install ecss-web-conf
```

Web-конфигуратор позволяет сделать управление системой более наглядным и комфортным. Установка web-конфигуратора не является обязательной, но рекомендуется.
Также при установке пакета **ecss-web-conf** автоматически устанавливается пакет **ecss-subsriber-portal-ui**. Приложение "Портал абонента" системы ECSS-10 позволяет абонентам системы самостоятельно управлять услугами, просматривать информацию по совершенным вызовам, активным конференциям, а также настраивать собственные IVR-скрипты для входящих вызовов. Описание работы веб-конфигуратора приведено в разделе "Портал абонента".

Вопросы ecss-web-conf	Ответы для усыы1	Пример
Хотите ли вы использовать стандартные настройки?	Yes (значение по умолчанию)	

Проверьте возможность подключение к web интерфейсу http://10.0.20.41, пользователь/пароль - admin/password, в настоящий момент не все функции будут доступны, но web интерфейс должен быть рабочий и логин успешный.

9.8.7 Настройка snpmd

Настройка SNMP описана здесь.

9.8.8 Настройка безопасности. SSH

Выполните настройку сервера SSH:

```
sudo nano /etc/ssh/
sshd_config
```

Укажите в конфигурационном файле порт и адрес, по которому можно обратиться к серверу:

Настройка ssh для ecss1(/etc/ssh/sshd_config)

```
# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Port 2000
#AddressFamily any
ListenAddress 10.0.10.10
#ListenAddress ::

<...>
```

Перезапустите ssh:

```
sudo systemctl restart ssh.service
```

9.8.9 Добавление услуг

Установка всех услуг указных в лицензии:

выполняется в CoCon

```
/cluster/storage/ds1/ss/install ds1@ecss1 ss_*
```

Результат:

```
/cluster/storage/ds1/ss/install ds1@ecss1 ss_*
Successfully installed: /var/lib/ecss/ss/ss_dnd.xml
Successfully installed: /var/lib/ecss/ss/ss_clir.xml
...
Successfully installed: /var/lib/ecss/ss/ss_chunt.xml
```

9.9 Первоначальная конфигурация системы с помощью web-конфигуратора

пример первоначальной конфигурации через Web интерефейс

9.9.1 Исходные данные

- Установка системы завершена;
- Система готова к дальнейшей настройке;
- Интерфейсы запущены.

Рекомендуется использовать последние доступные версии браузеров. Рекомендуемые к использованию браузеры: Яндекс, Chrome.

Для начала конфигурации системы необходимо зайти в web-конфигуратор по адресу http://<Ipadd SSW>.

Для определения и регистрации в системе планируются:

- Абоненты с номерами 42001, 42002 , 42003, 42004 ,42005, 42006, 42007, 42008, 42009, 42010;
- Транк в сторону шлюза.

9.9.2 Подготовка к работе

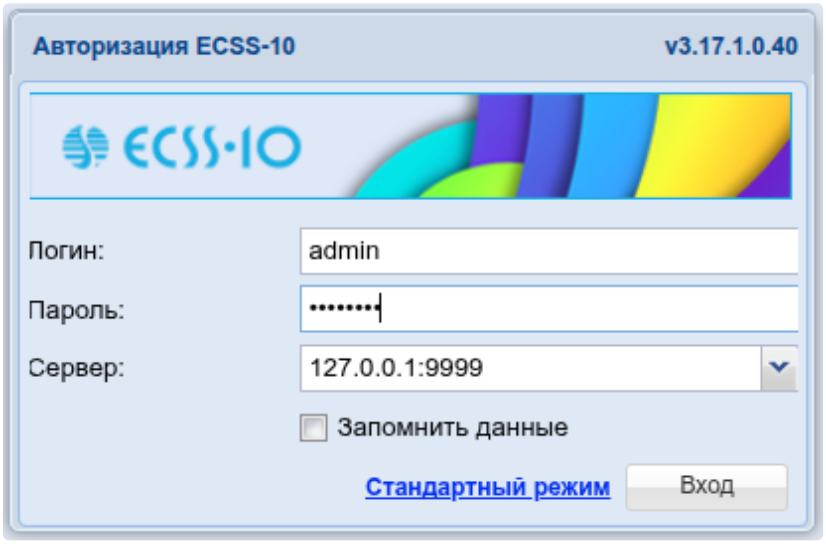


Рисунок 3 — Вход в web-конфигуратор (окно авторизации)

В окне авторизации необходимо ввести значения, определенные при установке web-конфигуратора.

> ⚠ Стандартные значения для авторизации:
> **Логин:** admin
> **Пароль:** password

После входа в систему будет видна основная рабочая область с иконками приложений, а также строка состояния с доступными опциями, в частности:

- 1 — выход из системы;
- 2 — выбор домена;
- 3 — выбор языка.

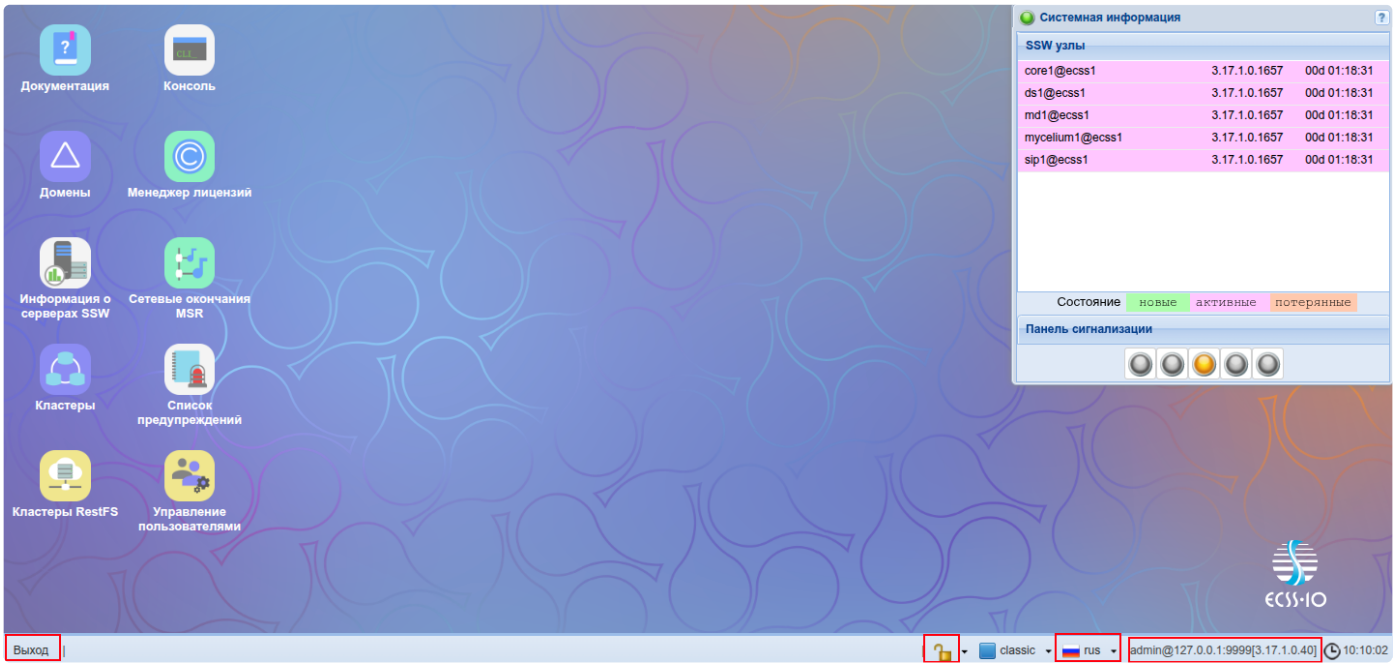


Рисунок 4 — Вид рабочей области web-конфигуратора

9.9.3 Создание учетной записи оператора

После авторизации для повышения безопасности при эксплуатации программного коммутатора рекомендуется создание учетных записей для операторов, а также сменить пароль для пользователя **admin**.

Для создания новой учетной записи оператора используется приложение "Управление пользователями" ("User manager"):

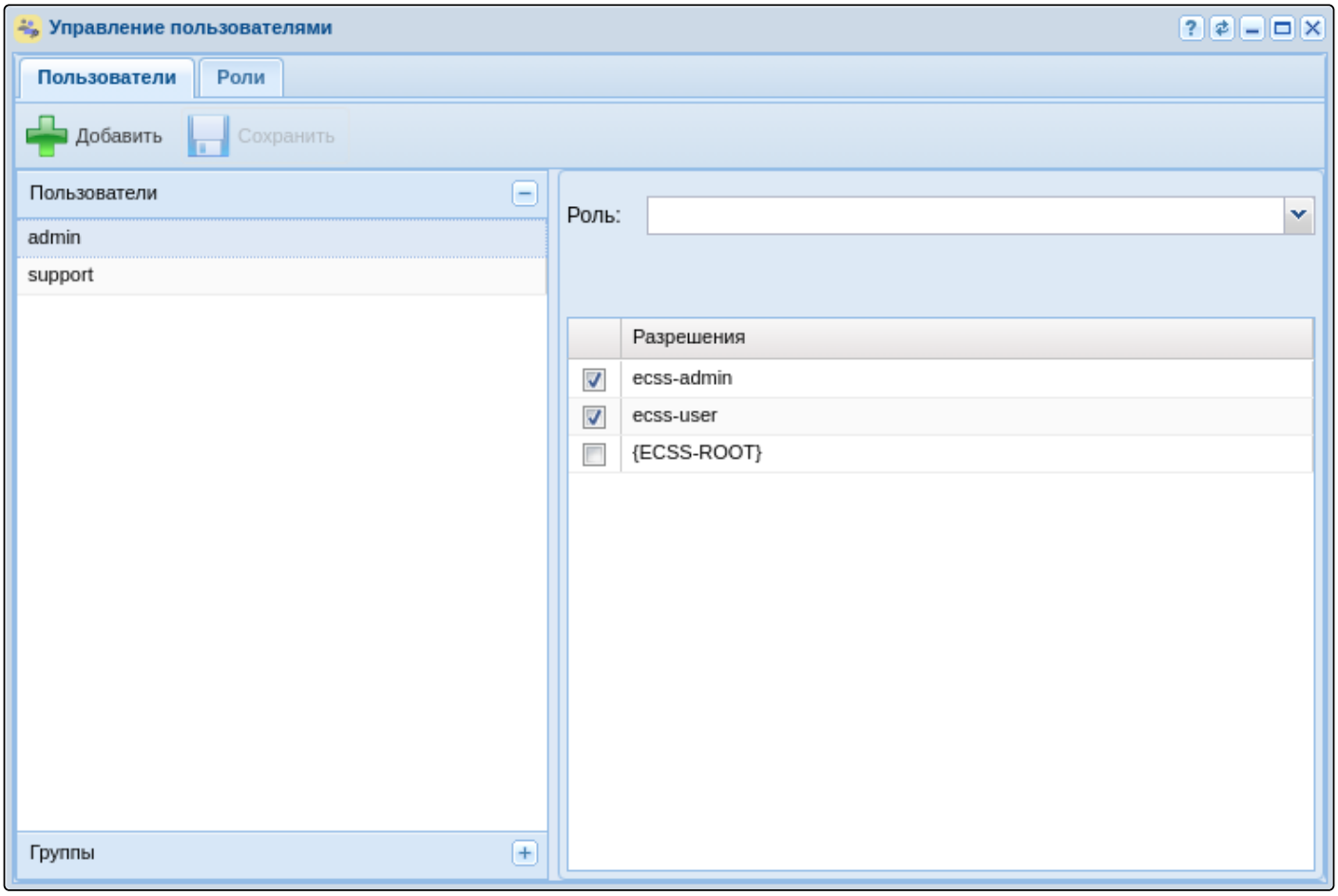


Рисунок 5 — Вид приложения "*Управление пользователями*"

Нажмите кнопку "*Добавить*"  . В открывшемся окне требуется определить новую учетную запись, для этого:

1. В поле "*Имя*" введите логин учетной записи, например "test";
2. В поле "*Пароль*" и "*Подтверждение*" введите пароль для пользователя, например "testpassword";
3. Определите уровень прав доступа для пользователя через выбор текущих разрешений или с помощью ролей, например **ecss-user**. Описание можно посмотреть в разделе "Управление пользователями".

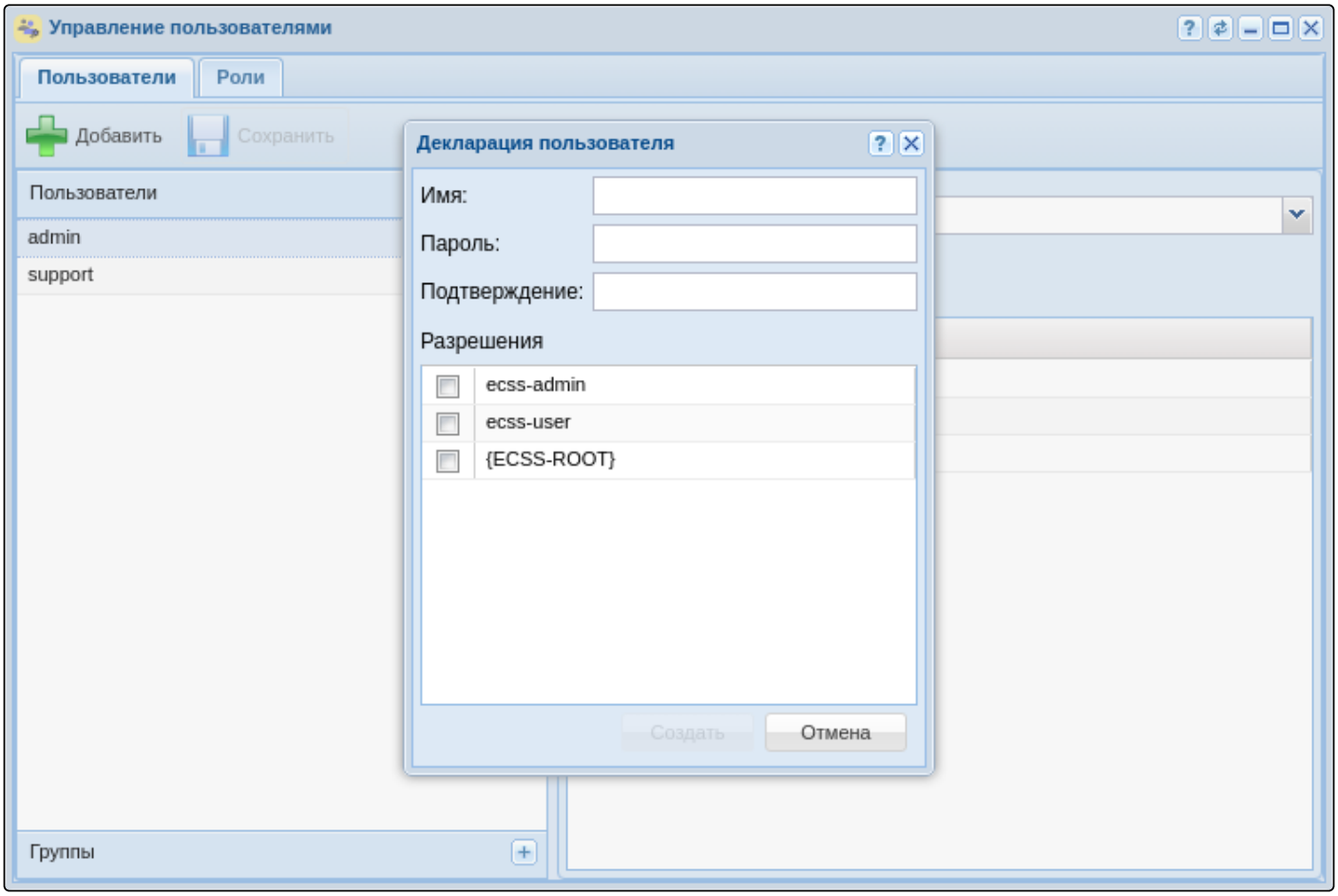


Рисунок 6 – Диалоговое окно создания учетной записи оператора

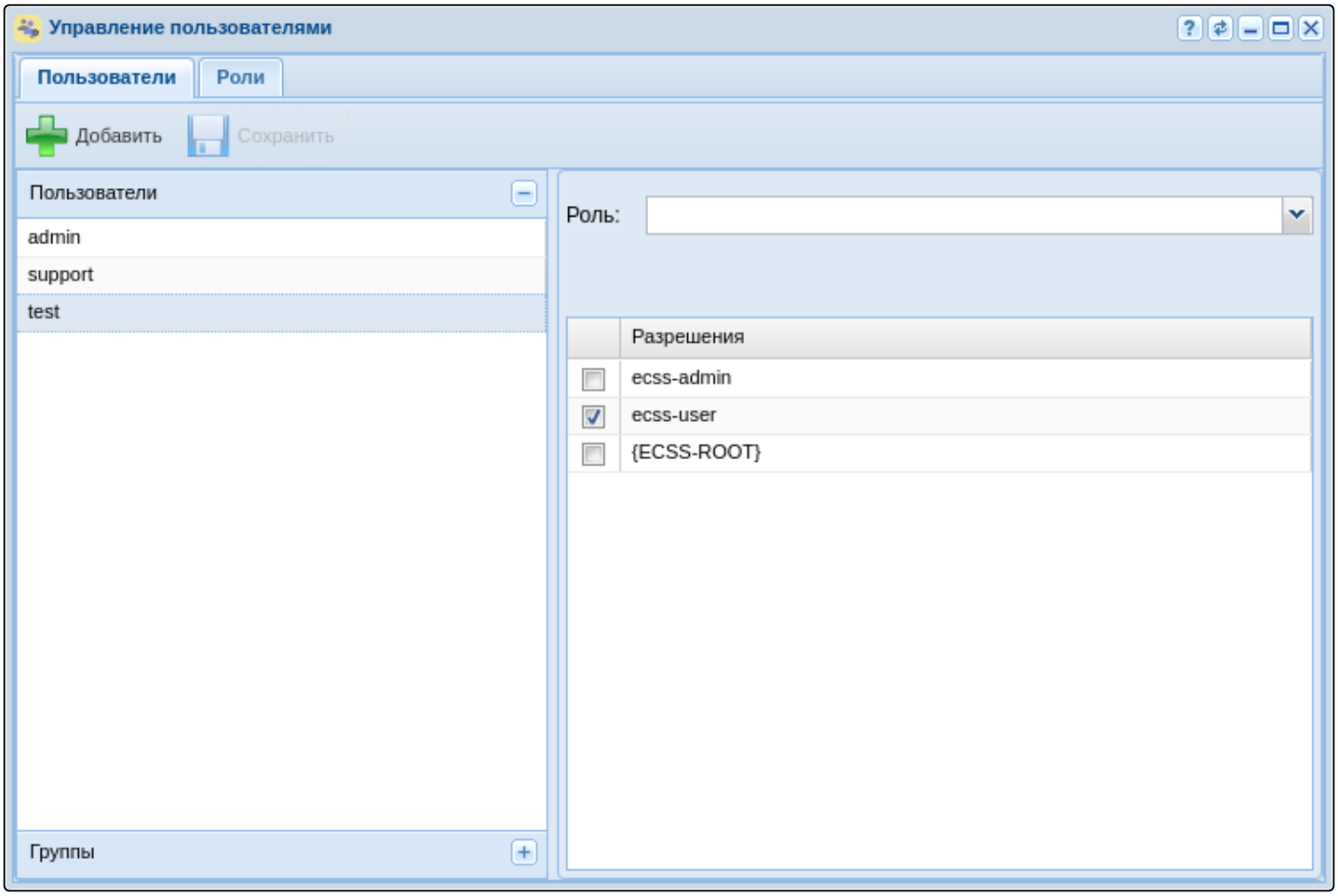


Рисунок 7 — Вид приложения с созданной учетной записью оператора

Для смены пароля рядом с названием пользователя нажмите на кнопку редактировать. В появившемся диалоговом окне введите:

1. Старый пароль (для пользователя "**admin**" по умолчанию паролем является "**password**");

2. Новый пароль;

3. Подтвердите новый пароль.

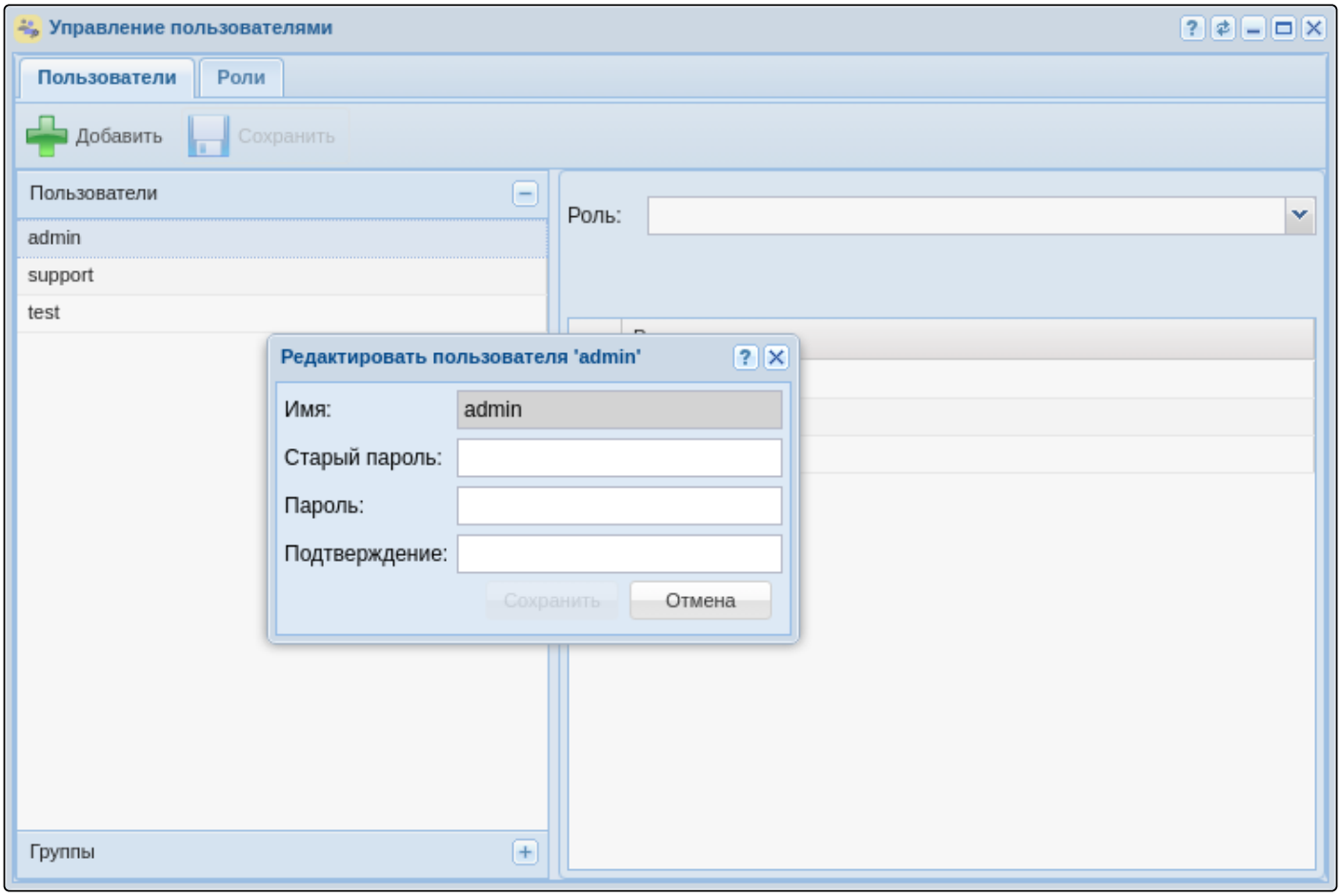


Рисунок 8 – Диалоговое окно редактирования учетной записи

9.9.4 Добавление медиа-ресурсов.

Для того чтобы начать работу с медиа-сервером сразу после его регистрации, необходимо активировать сетевые интерфейсы MSR. Приложение web-конфигуратора "Сетевые окончания MSR" ("MSR registrars") служит для настройки медиа-ресурсов.

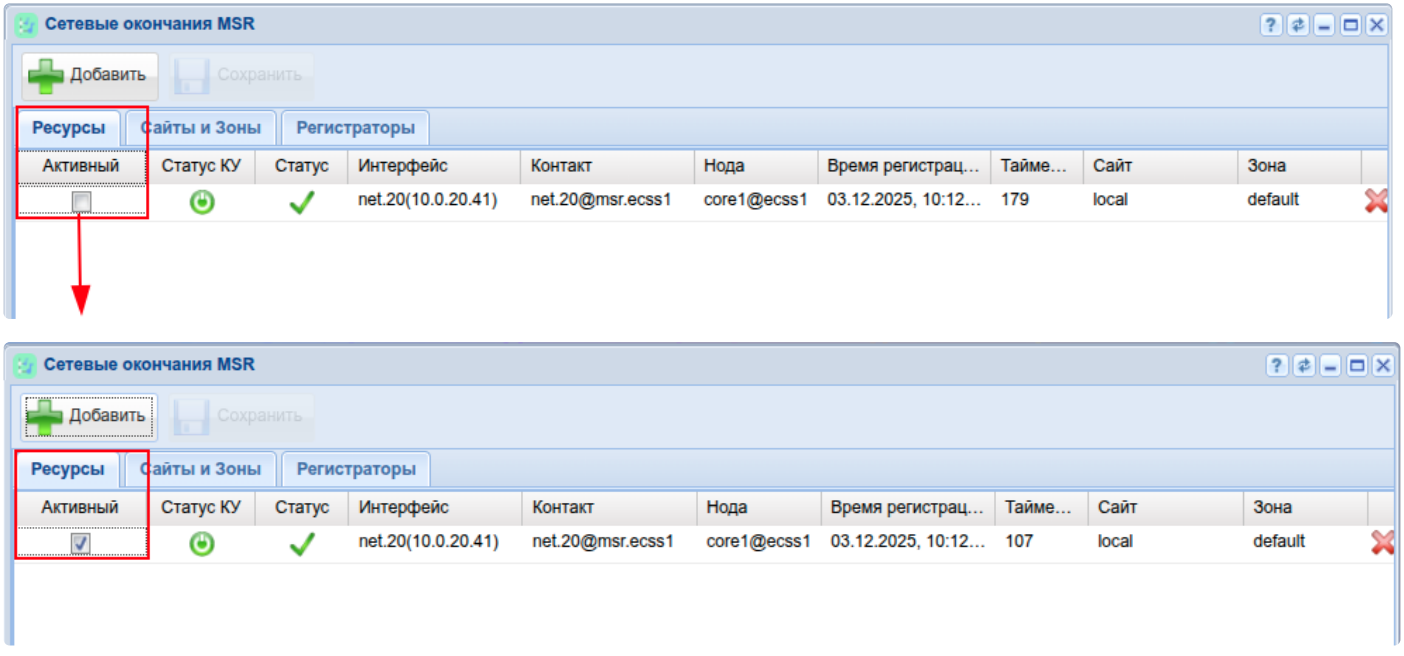
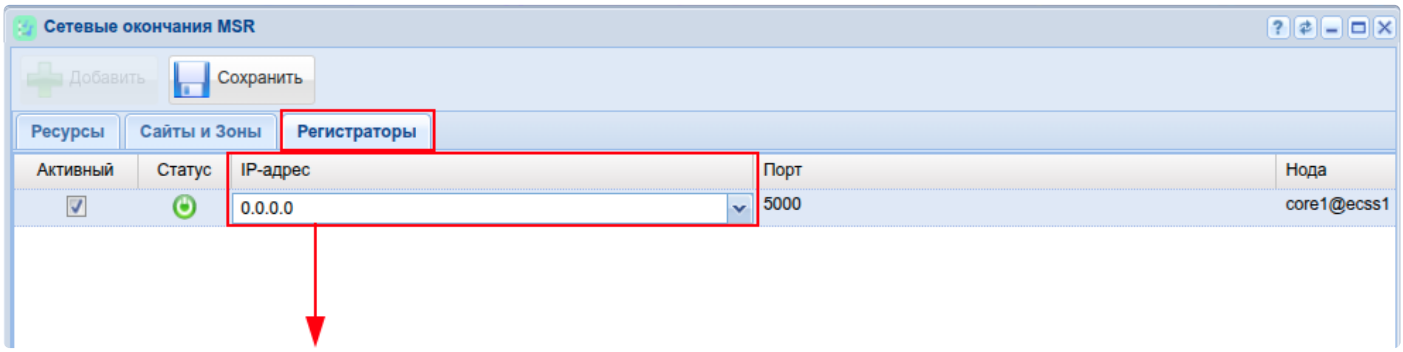
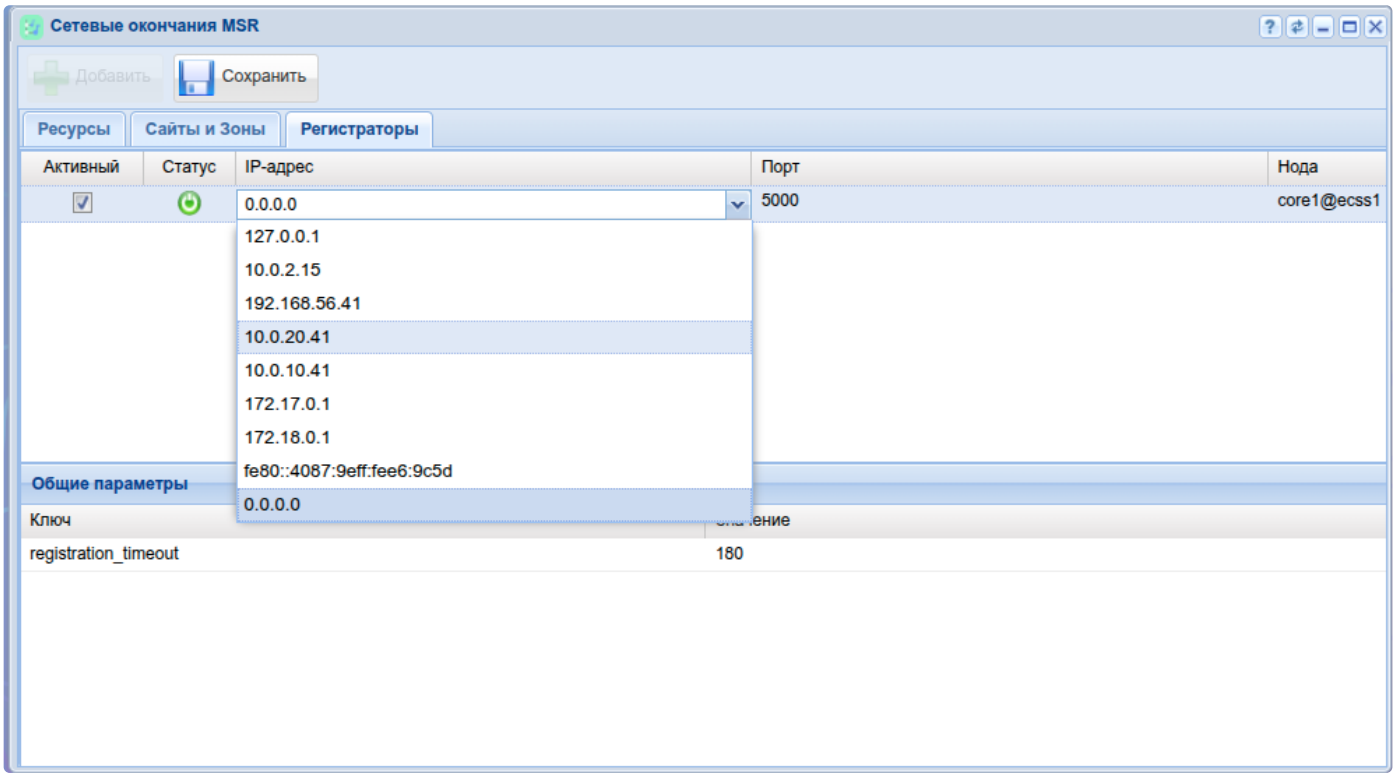


Рисунок 9 – Сетевые окончания MSR

Установить ipадрес для приема соединений от медиасервера, изменив значение 0.0.0.0 на реальные адреса MSR:





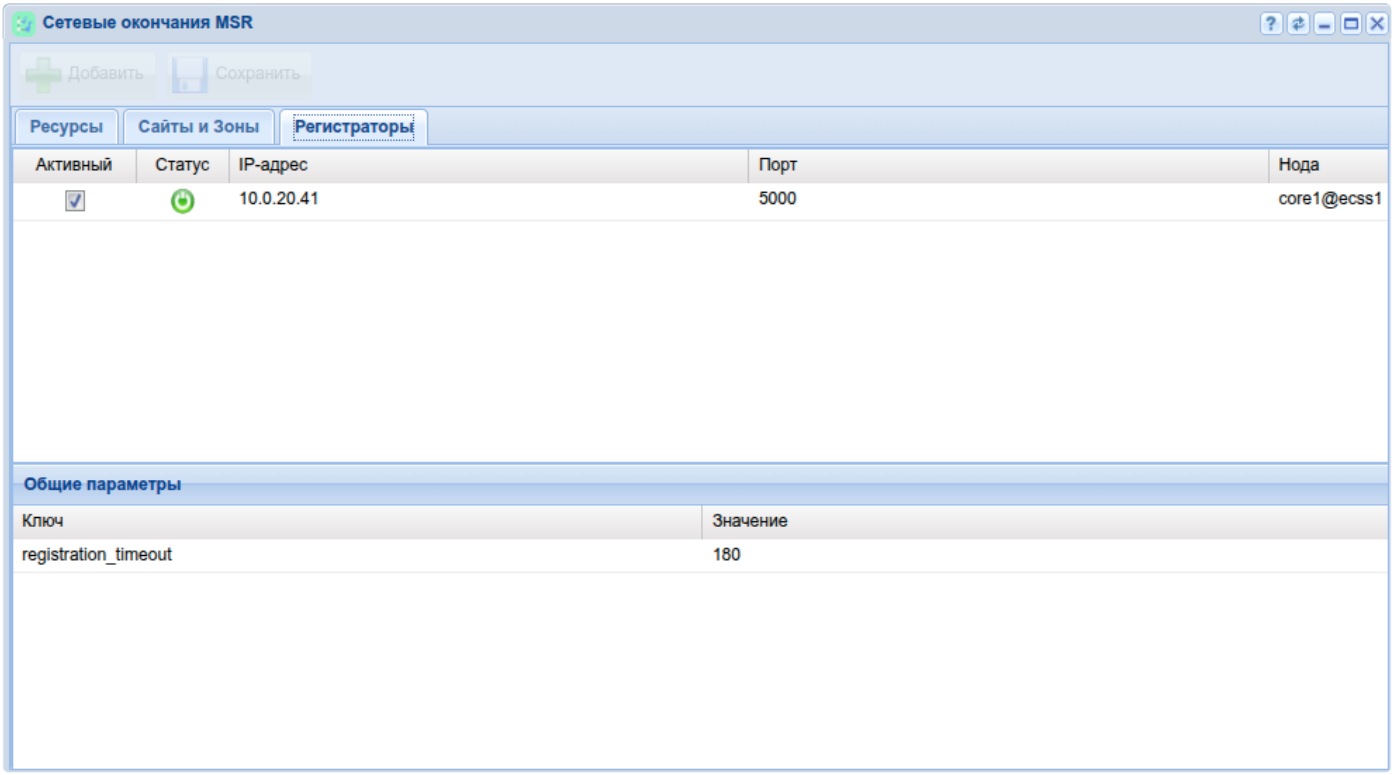


Рисунок 10 — Установка IP-add регистратора MSR

9.9.5 Создание IP-set (sip-транспорта) и присвоение его к домену

IP-set — совокупность одного или нескольких IP-адресов из одной подсети и списка слушающих UDP/TCP-портов, которые будут открыты SIP-адаптером на данных адресах.

Для настройки интерфейса необходимо открыть приложение "Кластеры ("Clusters").

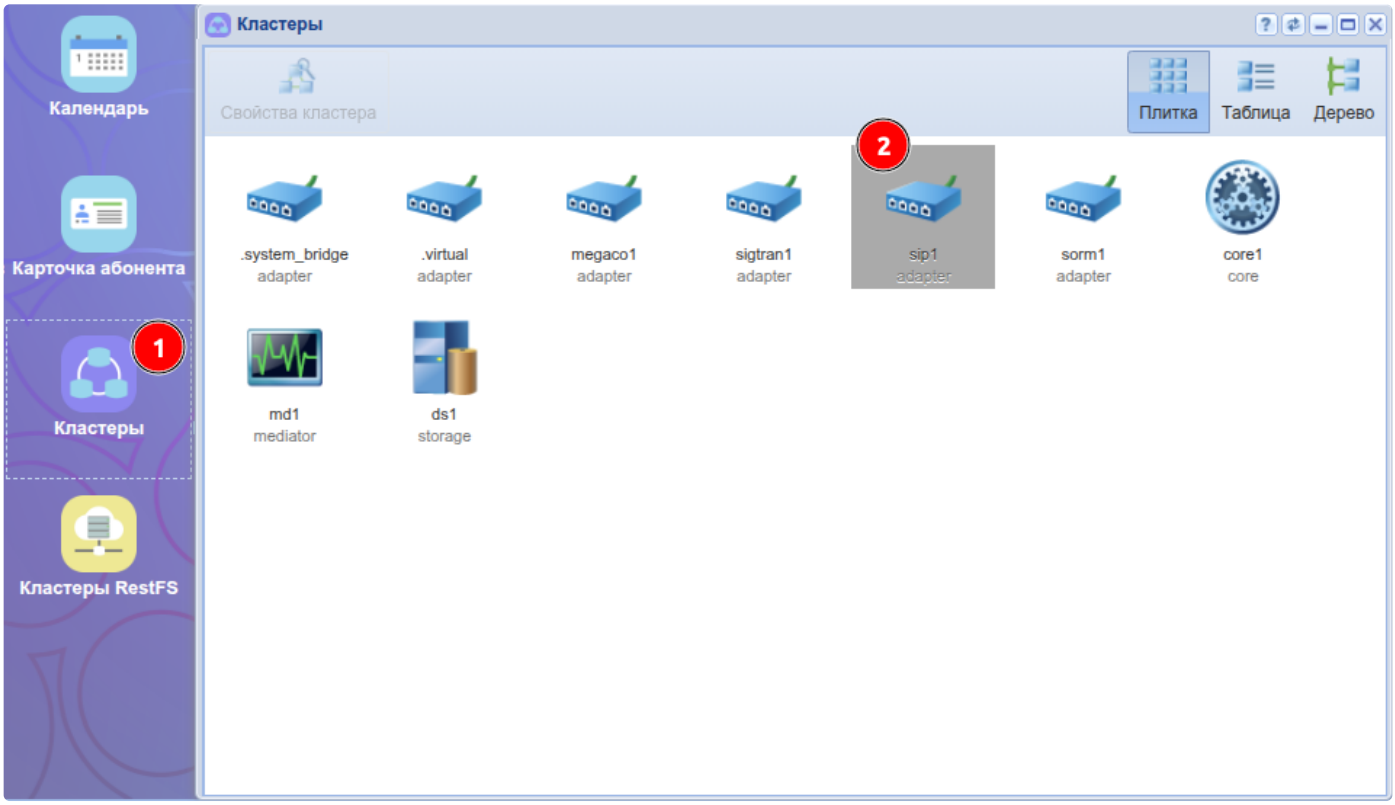


Рисунок 11 — Вид приложения "Кластеры"

Для создания новой группы IP-адресов (IP-set) выделите кластер SIP-адаптера "sip1" и нажмите на кнопку *"Свойства кластера"* (или нажмите на иконку кластера двойным щелчком левой кнопки мыши).

В появившемся диалоговом окне перейдите на вкладку *"Транспорт"*. Далее нажмите кнопку *"Добавить"*. Появится новая группа, для редактирования полей двойным щелчком кнопки мыши нажмите на нужное:

1. Переименуйте группу адресов (IP-set), например "test_set";

2. Задайте порт, по которому будет происходить обращение к домену, например 5060;

3. Разверните вновь созданную группу нажатием на треугольник слева от названия группы;

4. Определите адрес для ноды SIP-адаптера, согласно примеру конфигурации. Для примера 10.0.20.41:

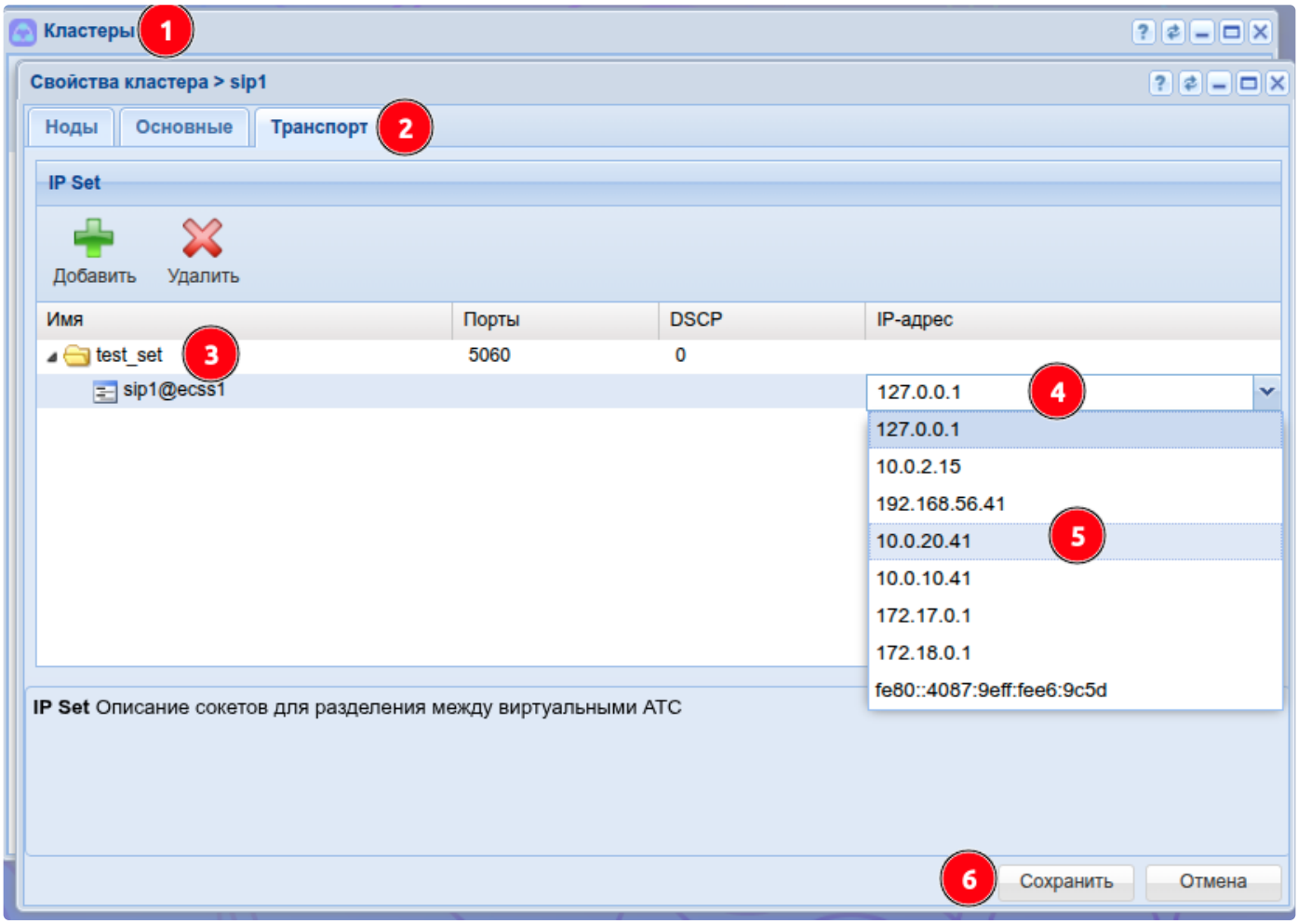


Рисунок 12 — Вид приложения "Кластеры/Транспорт"

Нажмите кнопку *"Сохранить"* для применения настроек.

9.9.6 Создание домена

Домен - Виртуальная PBX внутри нашего SSW. Таких виртуальных PBX может быть создано необходимое для проекта кол-во.

Для создания домена необходимо войти в приложение *"Домены" ("Domain")*. В открывшемся окне требуется создать домен, для этого:

1. Нажмите кнопку "*Добавить домен*":

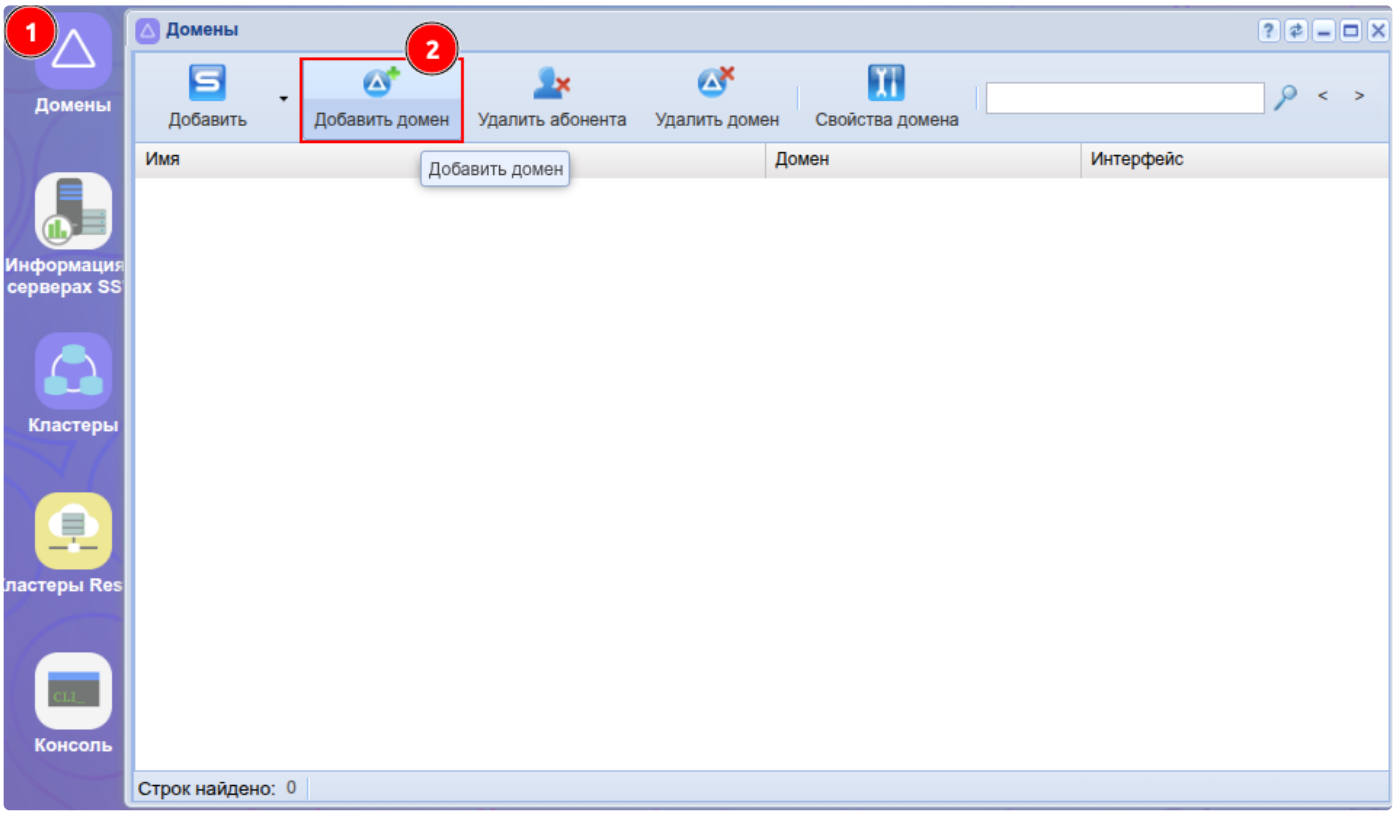


Рисунок 13 — Добавление домена в систему

2. В появившемся диалоговом окне доступны следующие настройки:

- *Имя (Name)* — индивидуальное имя виртуальной АТС;
- *Профиль услуг (SS profile)* — системный профиль дополнительных услуг. Этот профиль будет скопирован с тем же именем во вновь создаваемый домен и для всех услуг из этого профиля будет автоматически разрешён доступ через access-list;
- *IVR профиль (IVR profile)* — профиль IVR, заданный в приложении "Редактор IVR ограничений".

Введите название домена, например "test_domain";

3. Нажмите кнопку "*Ок*":

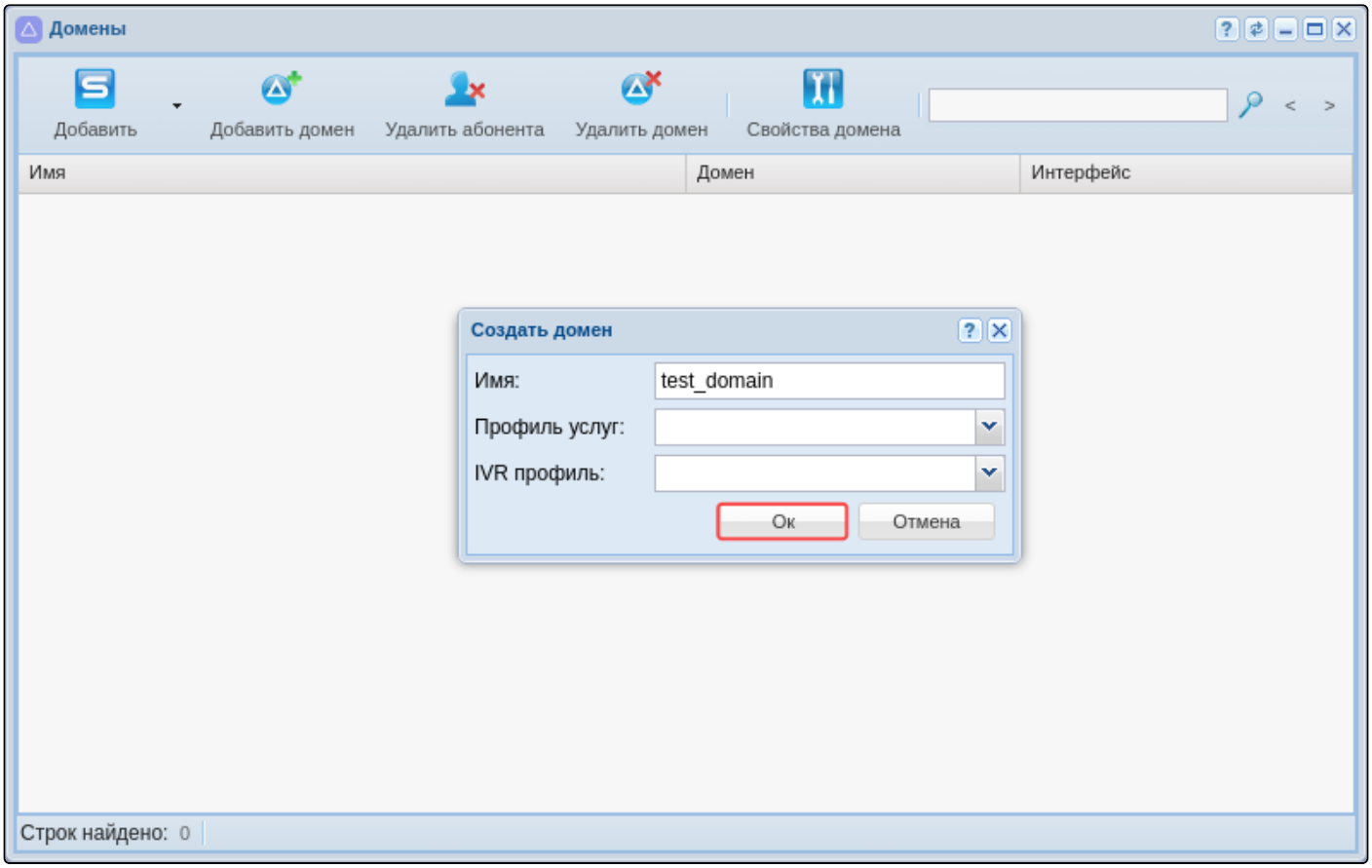


Рисунок 14 – Определение настроек нового домена

4. Нажмите кнопку *"Обновить"* в приложении.

Созданный домен будет отображаться в текущей конфигурации:

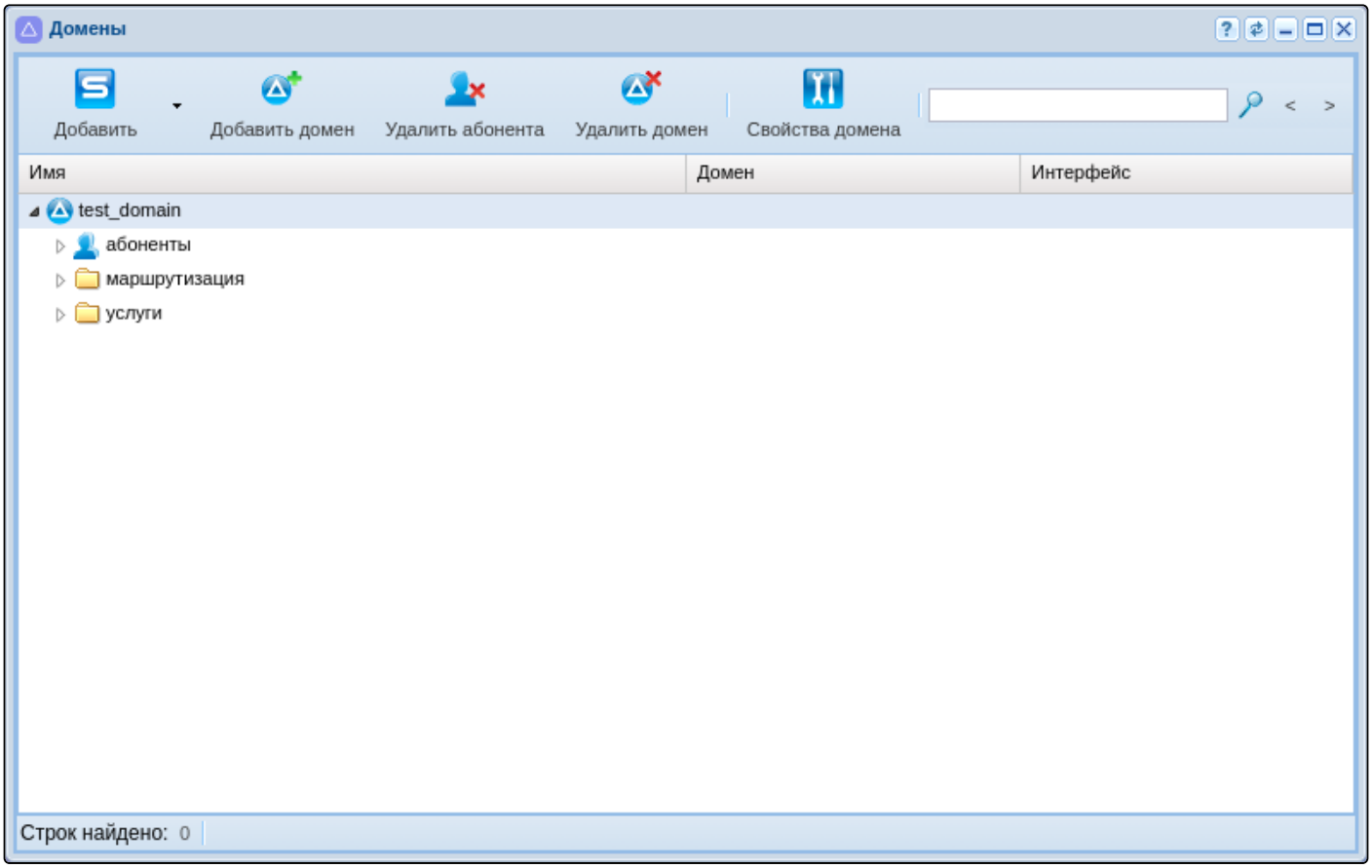


Рисунок 15 — Отображение созданного домена

Для редактирования текущего домена его необходимо выбрать в системе. Для перехода в домен воспользуйтесь опцией выбора домена (см. пункт 2 на рисунке "Вид рабочей области web-конфигуратора").

После выбора домена, согласно текущей конфигурации системы, станут доступны все приложения:

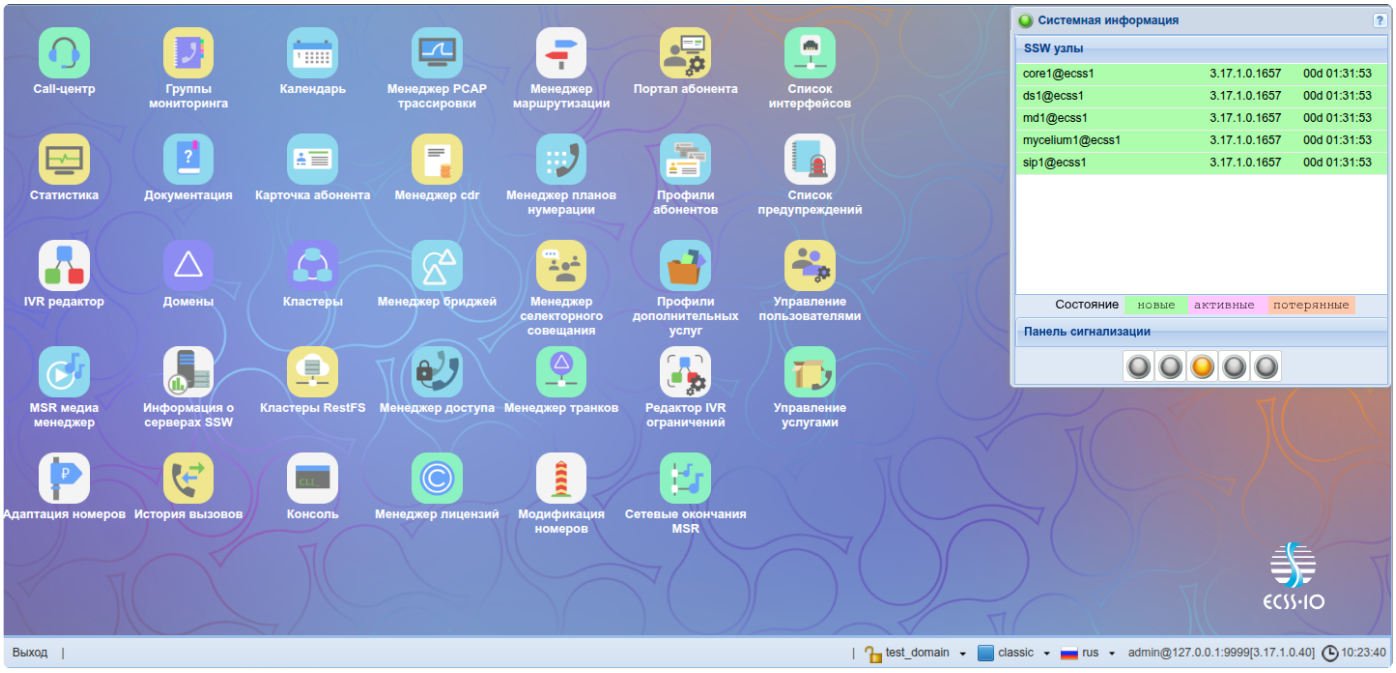


Рисунок 16 — Отображение приложений в текущей конфигурации системы

Для того чтобы **привязать группу адресов к домену**, необходимо вернуться в приложение *"Домены" ("Domain")*, выделить домен и перейти в настройки нажатием на кнопку *"Свойства домена"* или двойным нажатием левой кнопки мыши на домен.

В настройках списка откройте ветку "SIP", в ней "SIP транспорт", далее выберите в поле IP set созданную группу адресов. Нажмите кнопку *"Сохранить"* для применения настроек.

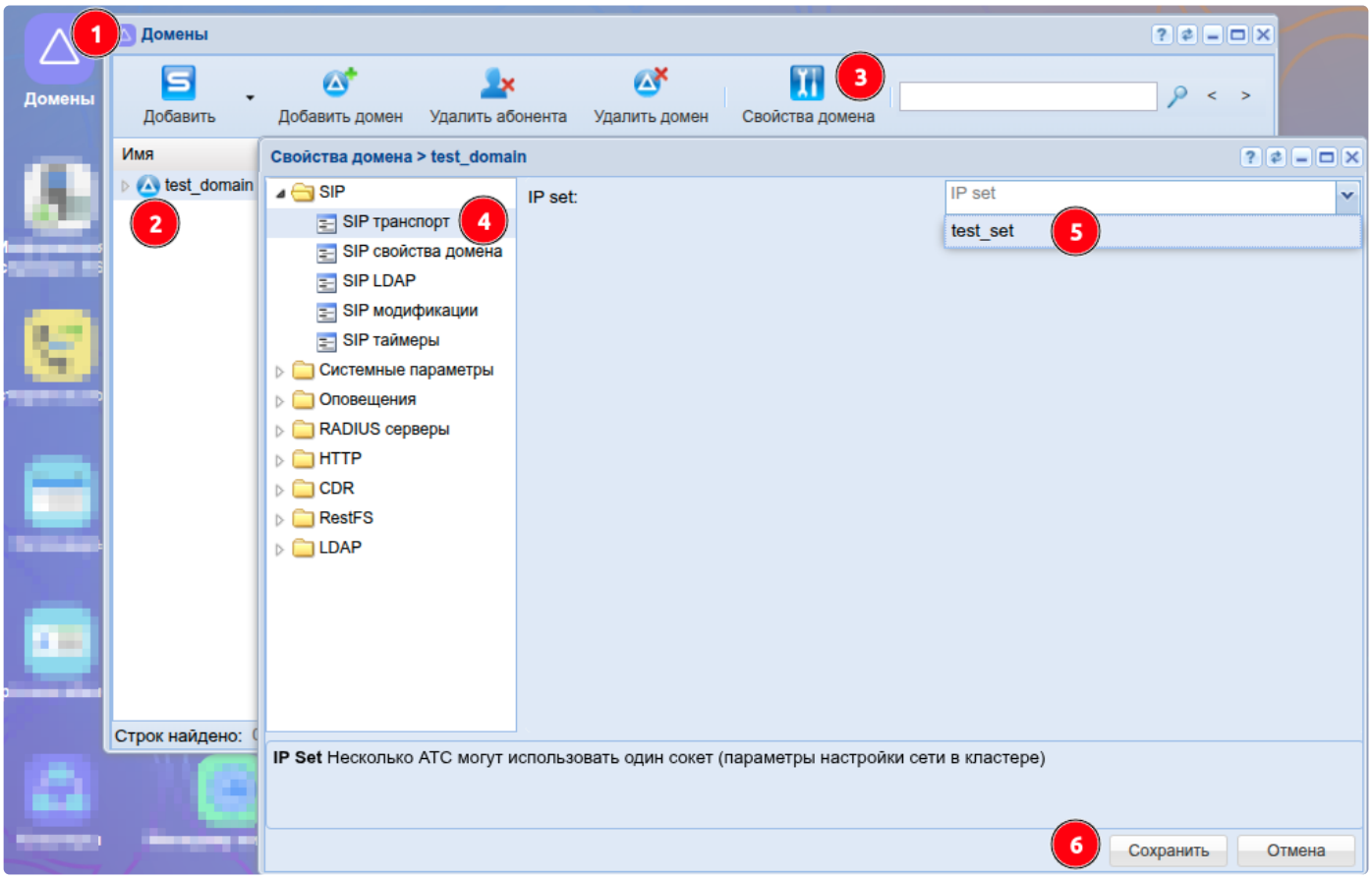


Рисунок 17 — Окно настройки для SIP транспорта

Выбрать требуемые услуги для созданного домена, которые будут использоваться (для примера для абонентского домена транковые услуги не востребованы , и наоборот для транзитного домена не нужно указывать услуги для абонента).

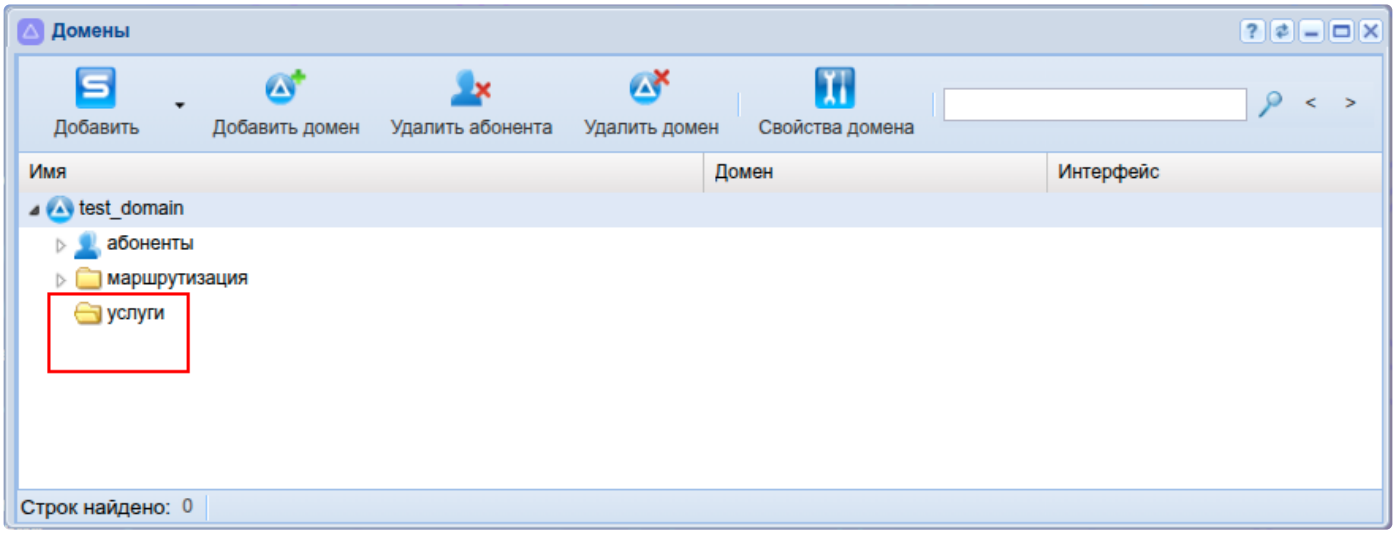


Рисунок 18 — После создания нового домена, услуги отсутствуют.

1. Выбрать приложение "Управление услугами"

2. Выбрать колонку "Установленные услуги" (услуги указанные в лицензии для SSW)
3. Выбрать необходимые для указанного домена услуги
4. Сохранить конфигурацию

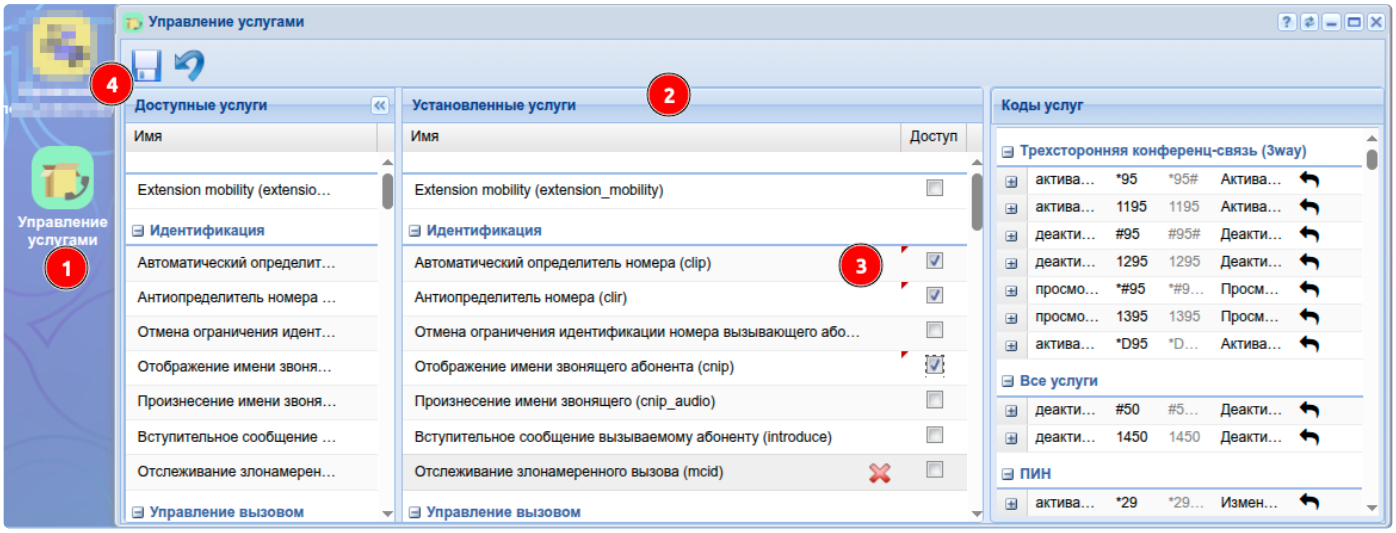


Рисунок 19 — Выбор услуг для абонентов / транков нового домена.

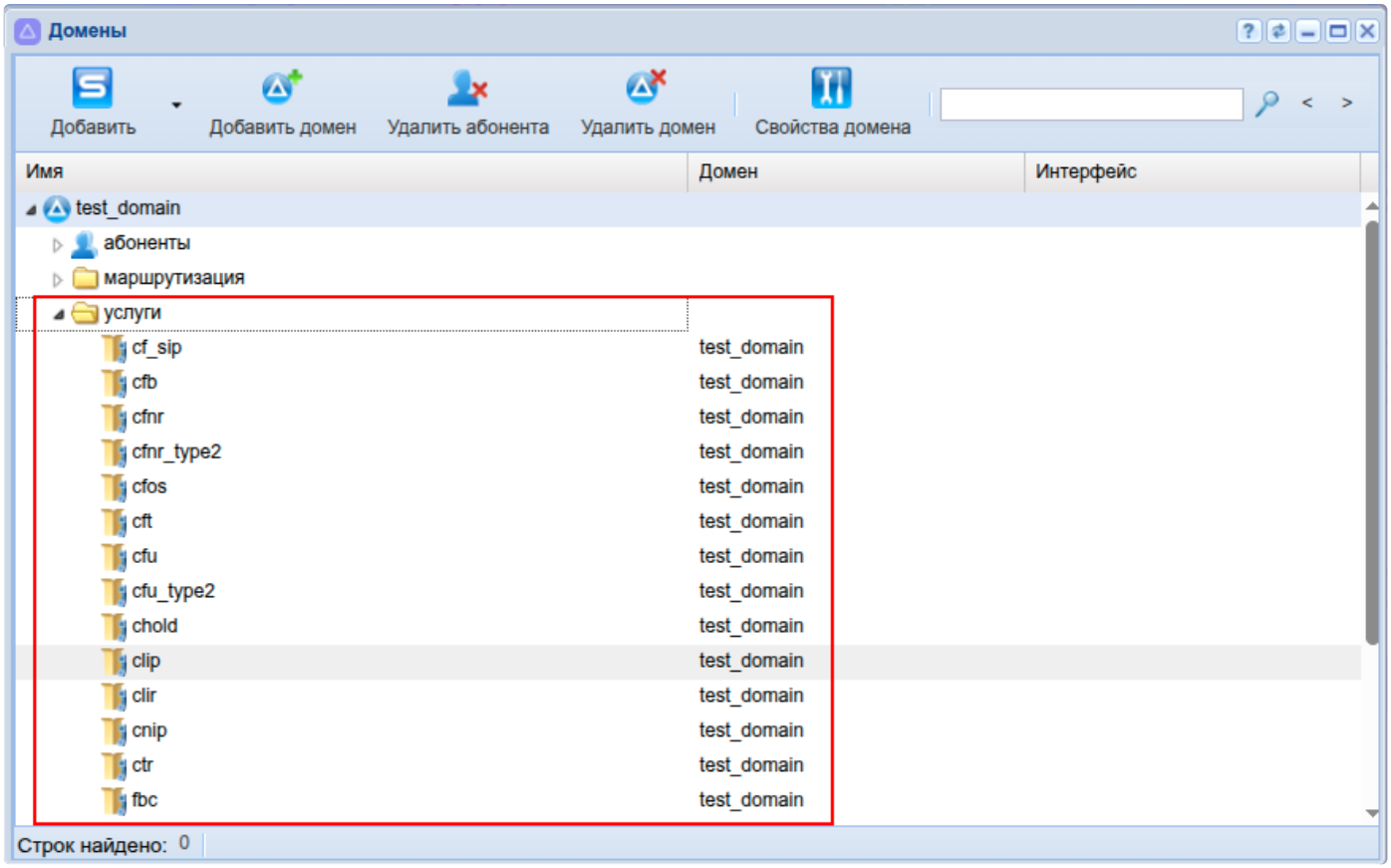


Рисунок 20 — Список услуг теперь доступных для абонентов нового домена.

9.9.7 Создание абонентов

Для создания и редактирования параметров абонентов в системе используется приложение "Карточка абонента" ("Subscriber card").

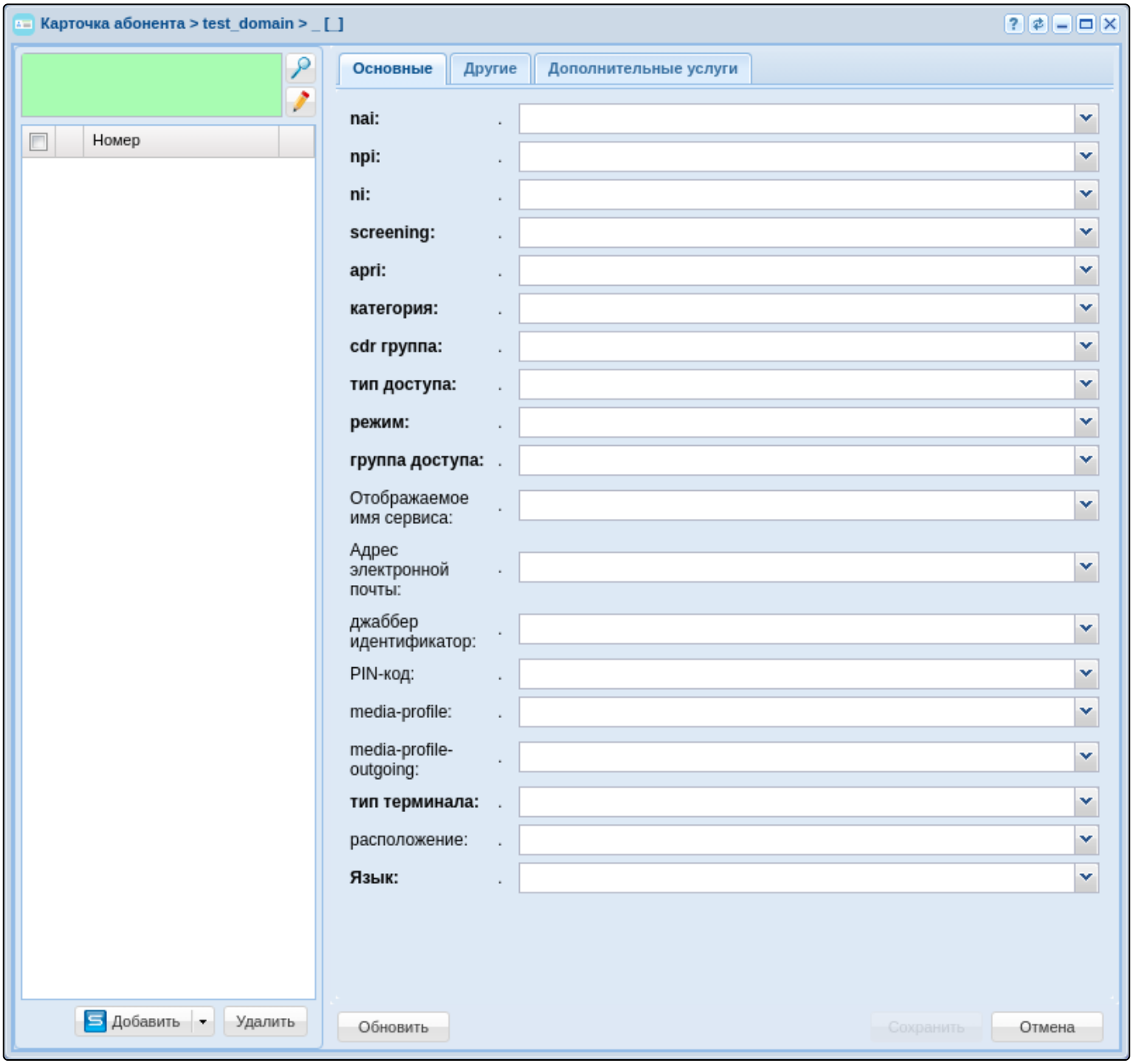


Рисунок 21 — Вид приложения "Карточка абонента"

В конфигурации возможно создание SIP-абонентов и виртуальных абонентов.

Для пользователей, имеющих физическое окончание, используется функционал SIP-абонента, в то время как виртуальный абонент используется тогда, когда нужна функциональность без физических окончаний. Например, номер для обращения к ivr-скрипту.

Для создания новых абонентов нужно нажать на кнопку *"Добавить"*.

В появившемся диалоговом окне укажите следующие параметры:

- *Имя интерфейса* — номер или группа номеров, который присваивается абоненту, строка, например {42001-42010} или abcd;
- *Контекст* — контекст маршрутизации, выбираем тот, который создали, например "test_name", или по умолчанию - "default_routing";
- Группа интерфейсов — имя группы объединяющей нескольких абонентов, для примера "sip";
- *Алиас как пользователь* — настройка, которая связывает номер сущности, алиас и пользователя одним и тем же названием, в примере настройка активирована;

- *Авторизация* — порядок проверки подлинности прав пользователя на доступ к данным, в примере используется"register";
- *Логин* — использую настройку номер как логин для простоты;
- *Пароль* — можно задать или использовать сгенерированный системой.

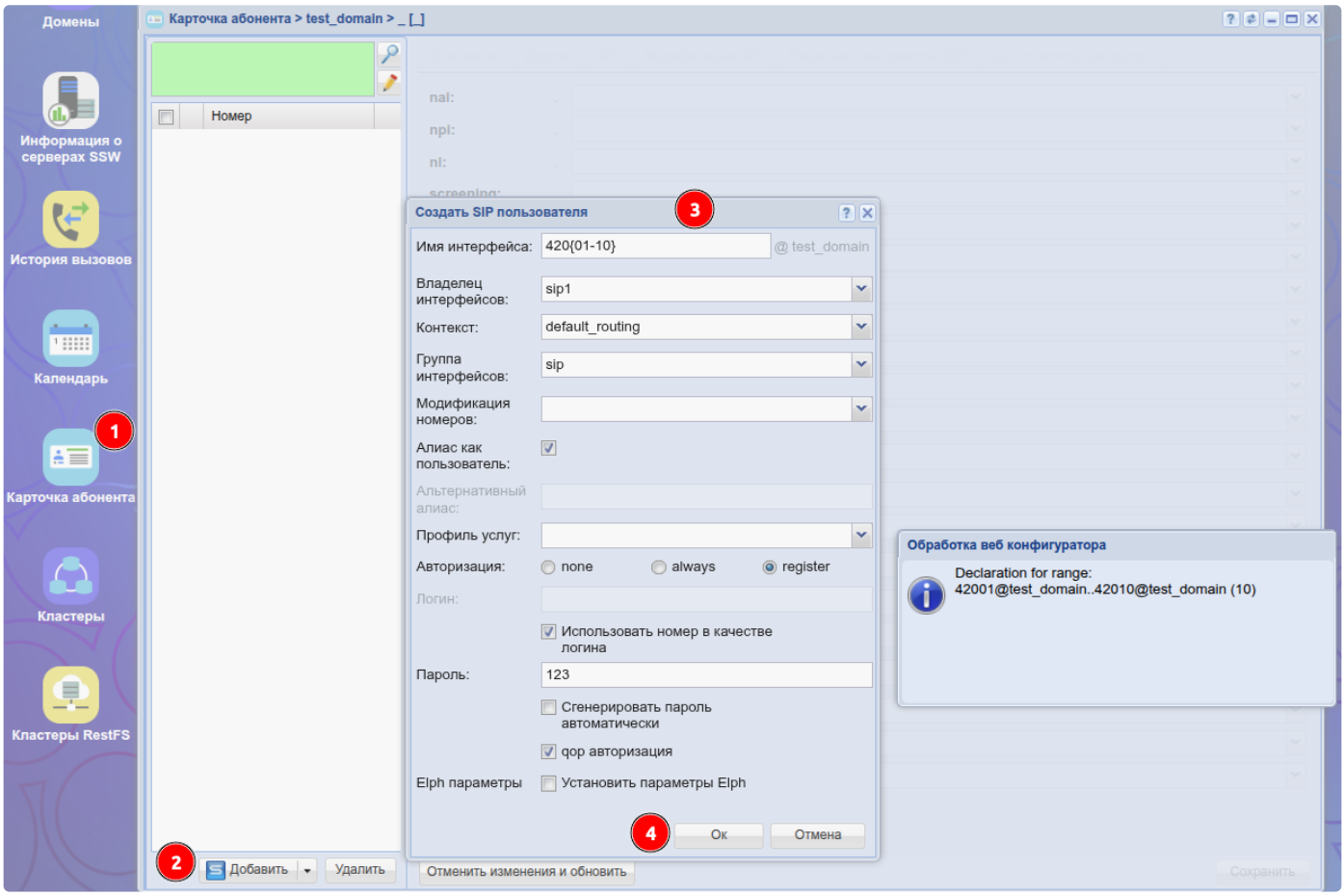


Рисунок 22 — Пример создания 10 абонентов в домене "test_domain"

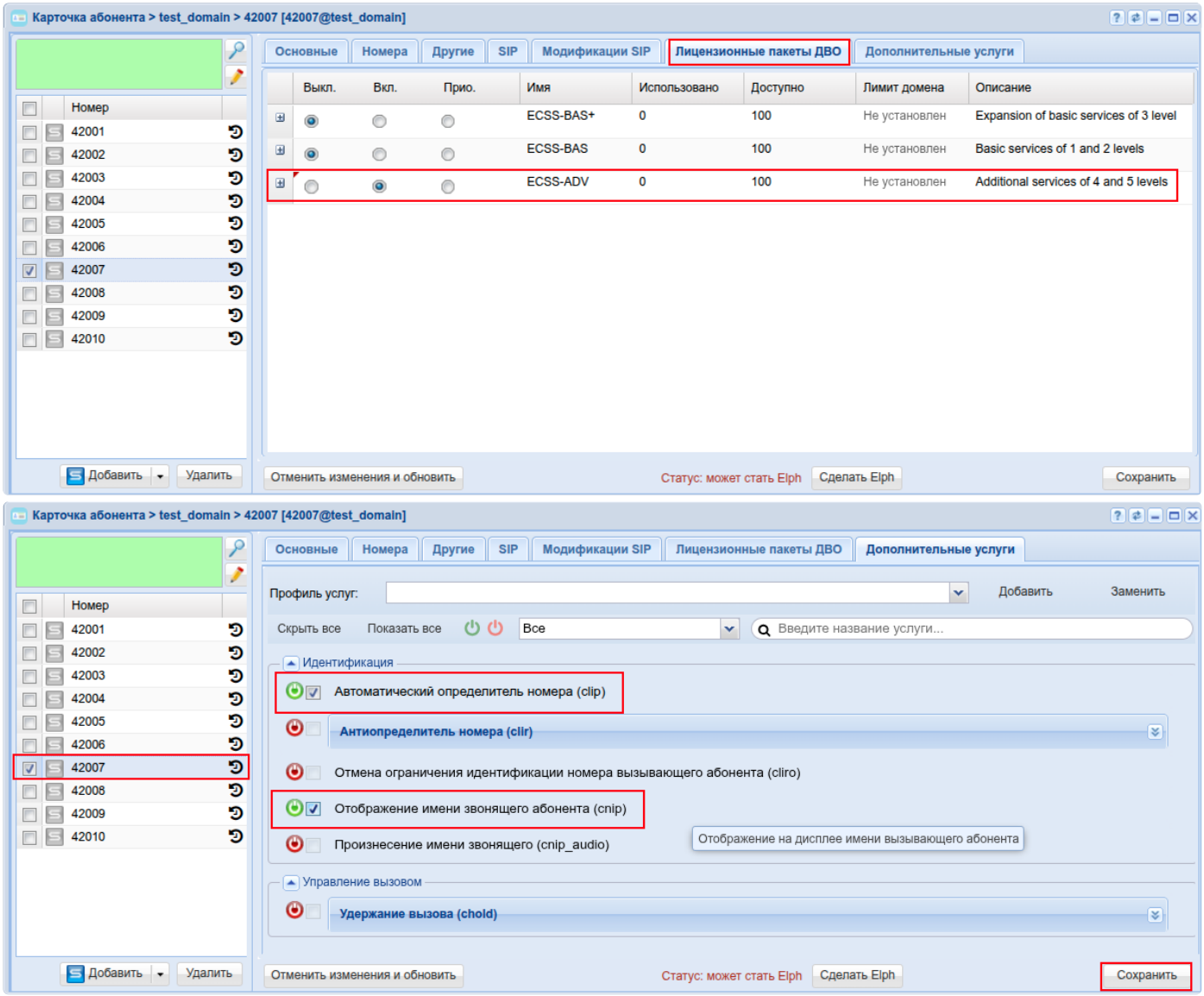


Рисунок 23 – Пример добавления услуг для абонента

9.9.8 Создание и применение контекстов маршрутизации для домена

Маршрутизация отвечает за поиск номера и последующую адресацию вызова. Для корректной работы системы должен быть настроен как минимум один контекст маршрутизации.

Настройка маршрутизации осуществляется в приложении "Менеджер маршрутизации" ("Routing manager") .

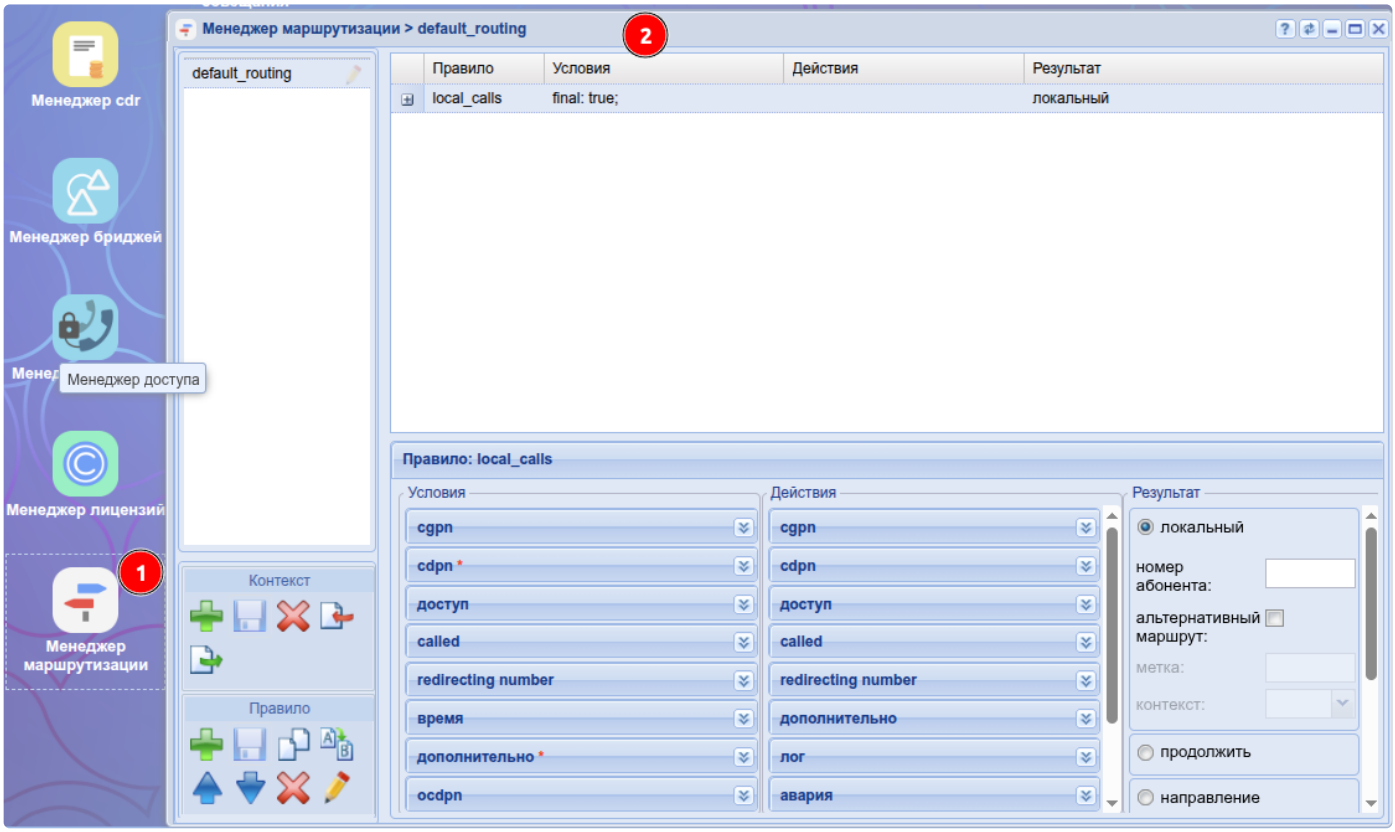


Рисунок 24 — Вид приложения "Менеджер маршрутизации"

В качестве примера создадим контекст, а затем несколько правил в нём, для этого:

1. В левой части окна в разделе "Контекст" нажмите на кнопку *"Создать контекст"*  ;
2. В открывшемся диалоговом окне обозначьте имя контекста:

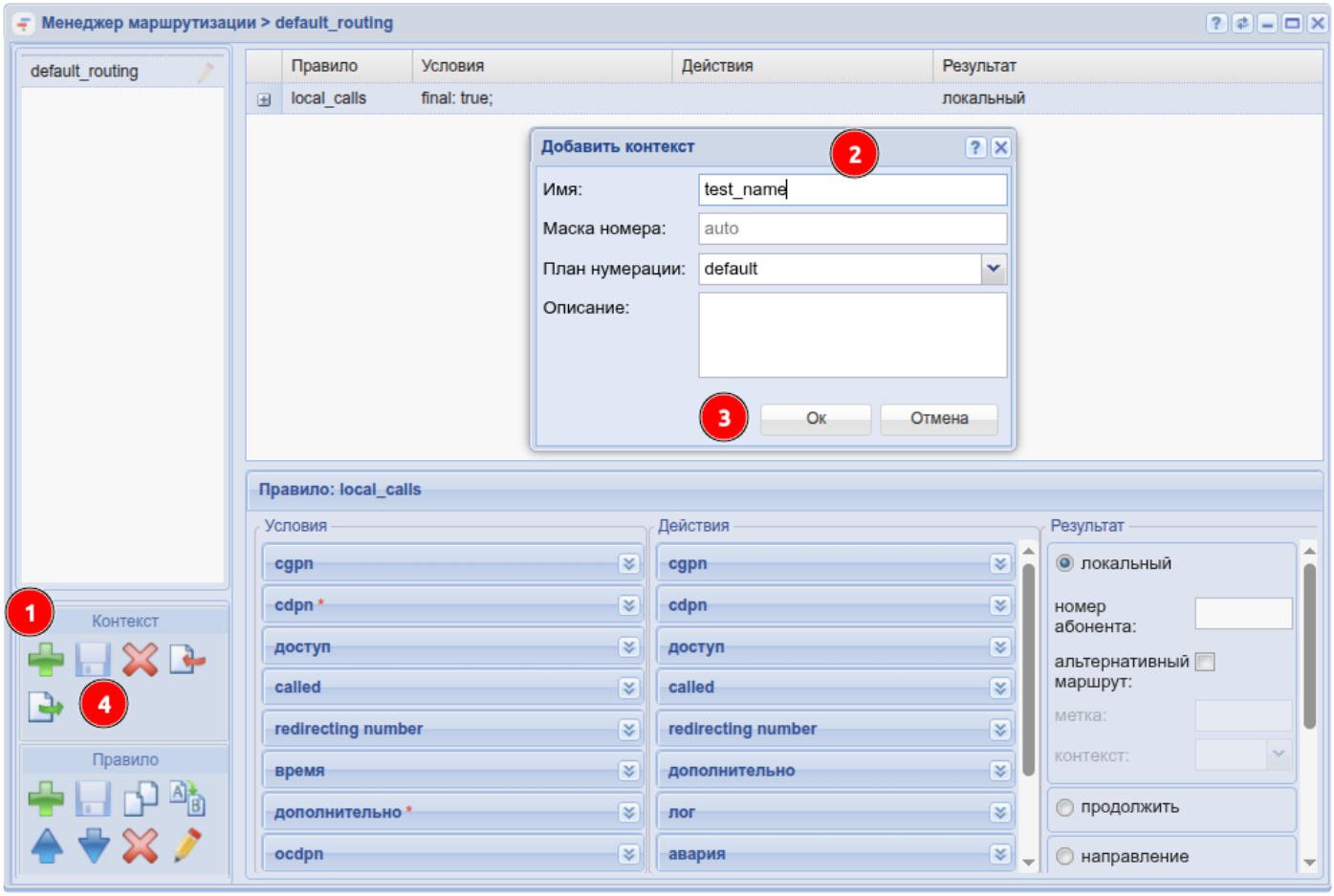


Рисунок 25 — Вид приложения "Менеджер маршрутизации" добавление нового контекста маршрутизации

3. Нажмите *"Сохранить контекст"*  ;

Создадим 4 правила в данном контексте:

- rule1 — правило для выхода на транк TAU-72;
- rule2 — правило локальной маршрутизации для номеров 42001-42005;
- rule3 — правило для выхода в ivr;
- rule4 — правило-исключение.

Для создания нового правила нужно выделить созданный контекст и нажать кнопку *"Создать правило"*. В появившемся окне нужно ввести название правила. Затем сохранить вновь созданные правила.

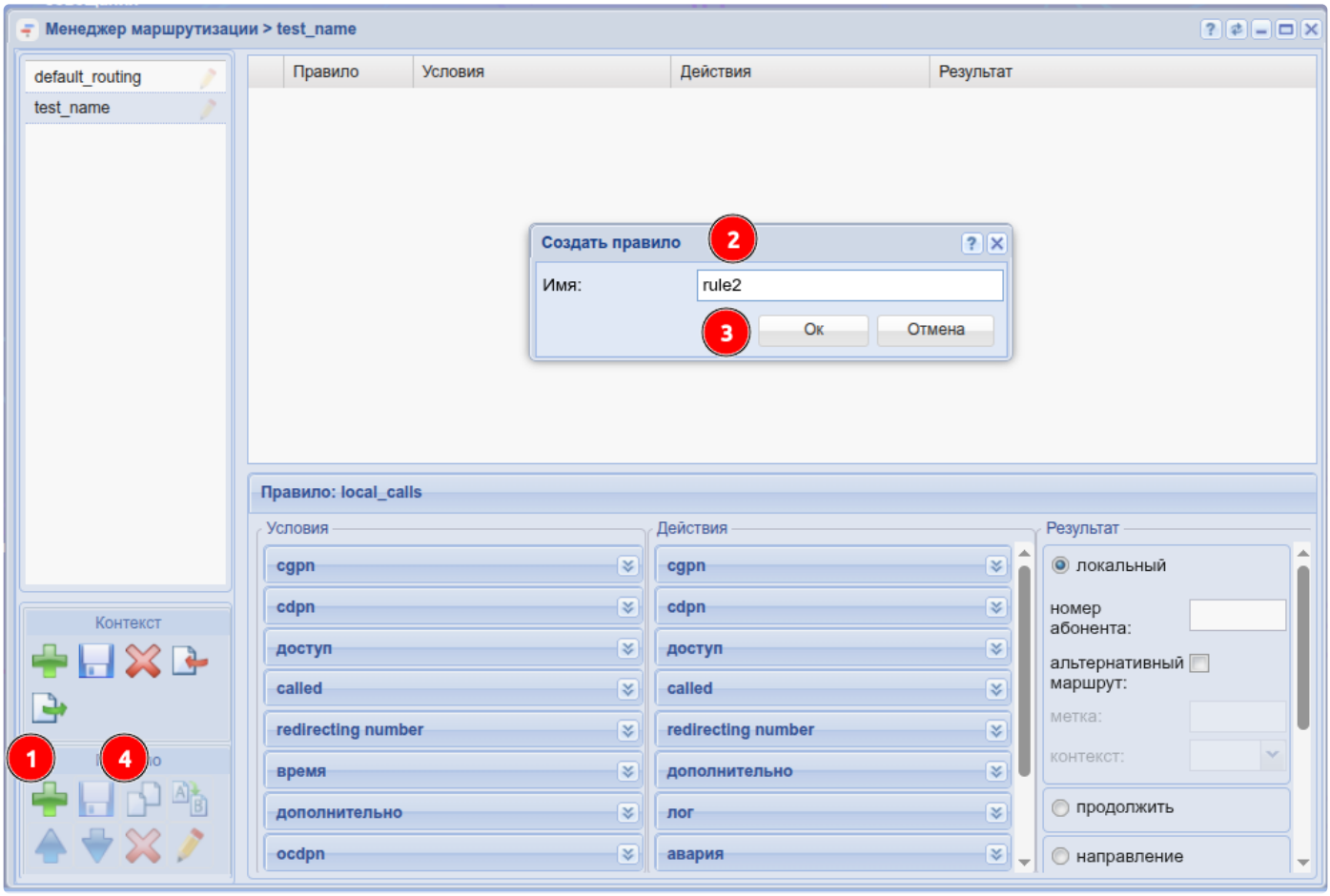


Рисунок 26 – Создание правила контекста маршрутизации

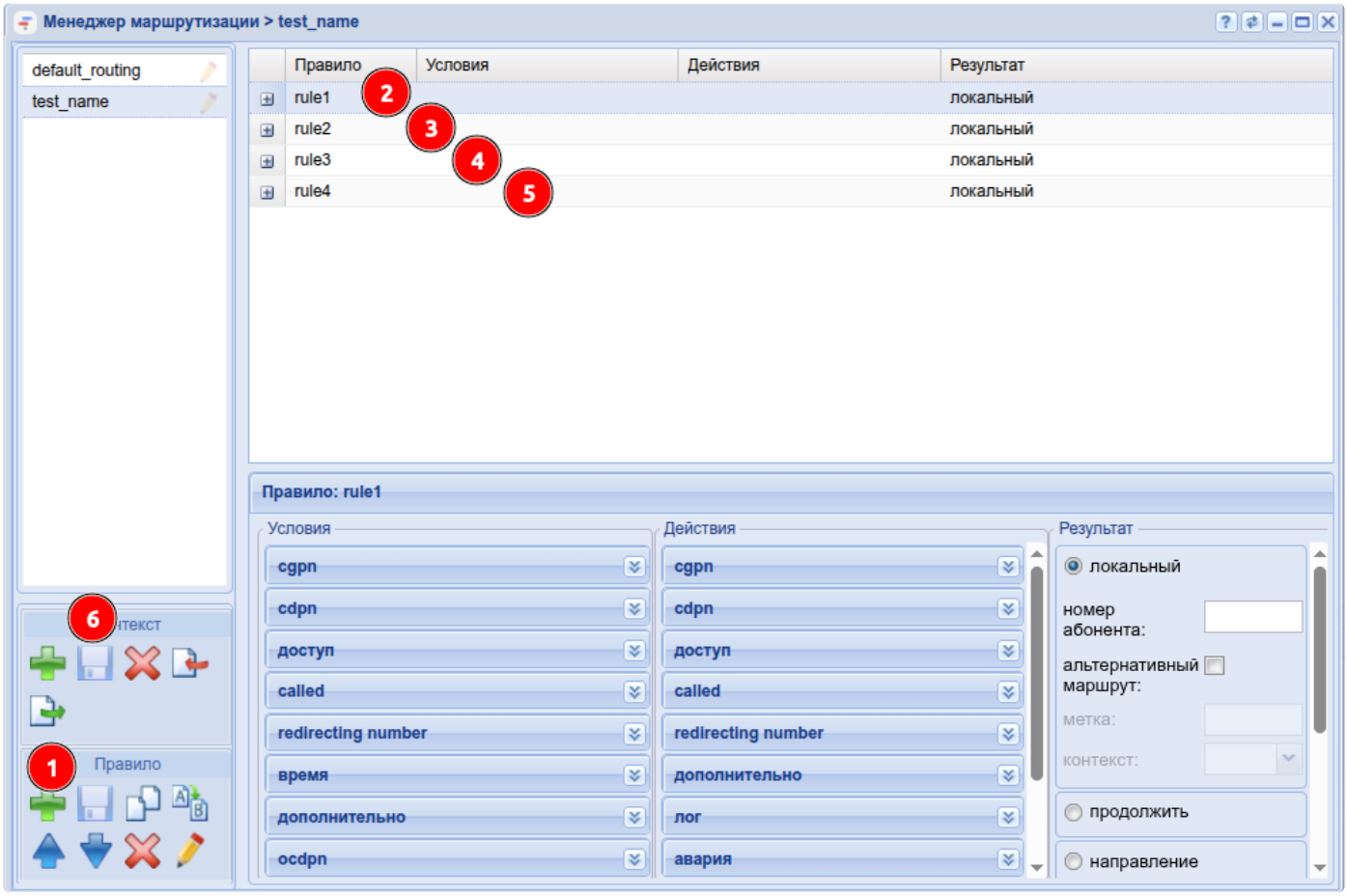


Рисунок 27 — Определение правил

На данный момент не определен транк, чтобы можно было сослаться на него в правиле, однако можно задать номера, по которым будет производится выборка.

Нажав на правило rule1, перейдем в нижнюю часть экрана, в которой располагаются области для редактирования контекста маршрутизации. Условно определим, что выборка для выхода на транк будет проводиться по признакам номера вызываемого абонента (CDPN), а номера в транке должны начинаться с цифры 4.

Функционально контекст маршрутизации делится на три части:

- Условия ("Condition") — раздел, определяющий выражения для выборки по предложенным признакам;
- Действия ("Action") — раздел, выполняющий преобразование признаков номеров к конкретному значению;
- Результат ("Result") — раздел, завершающий маршрутизацию и определяющий её результат.

rule1: Для выхода на транк нужно правильно отредактировать каждую часть:

- В разделе условий перейти на вкладку CDPN. Ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера с 42006 по 42007, нужно ввести условие: 420(6-7);
- В разделе действий перейти на вкладку CDPN. Ввести маску для изменения номера в поле "Номер". Например, чтобы добавить число 4 перед номером, нужно заполнить поле следующим выражением: 4{1,2,3,4,5};
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить транк в системе, поэтому к настройке данного правила вернемся чуть позже.

Таким же образом нужно настроить оставшиеся правила.

Для rule2:

- В разделе условий, во вкладке CDPN ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера со 42001 по 42005, ввести условие 420(1-5);
- В разделе результата определить результат как локальный (т.е. локальная маршрутизация);
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Для rule 3 допустим, что абоненты с номерами 108, 109 и 110 попадают в ivr-скрипт–информатор, прежде чем звонить далее.

- В разделе условий, во вкладке CGPN ввести номера телефонов, по которым будет выборка в поле "Номер", например 420(10,08-09).
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить в системе ivr-скрипт, поэтому к настройке данного правила вернемся чуть позже.

Для rule 4 мы определяем правило-исключение — это правило, которое работает в случае, если любые другие правила не срабатывают.

- По умолчанию данное правило создается в системе в контексте default_routing. В условиях на вызываемый номер прописывается символ %, а результатом является локальная маршрутизация, однако, если мы назначаем абоненту контекст, отличный от default_routing, в конце *рекомендуется* создавать это правило.

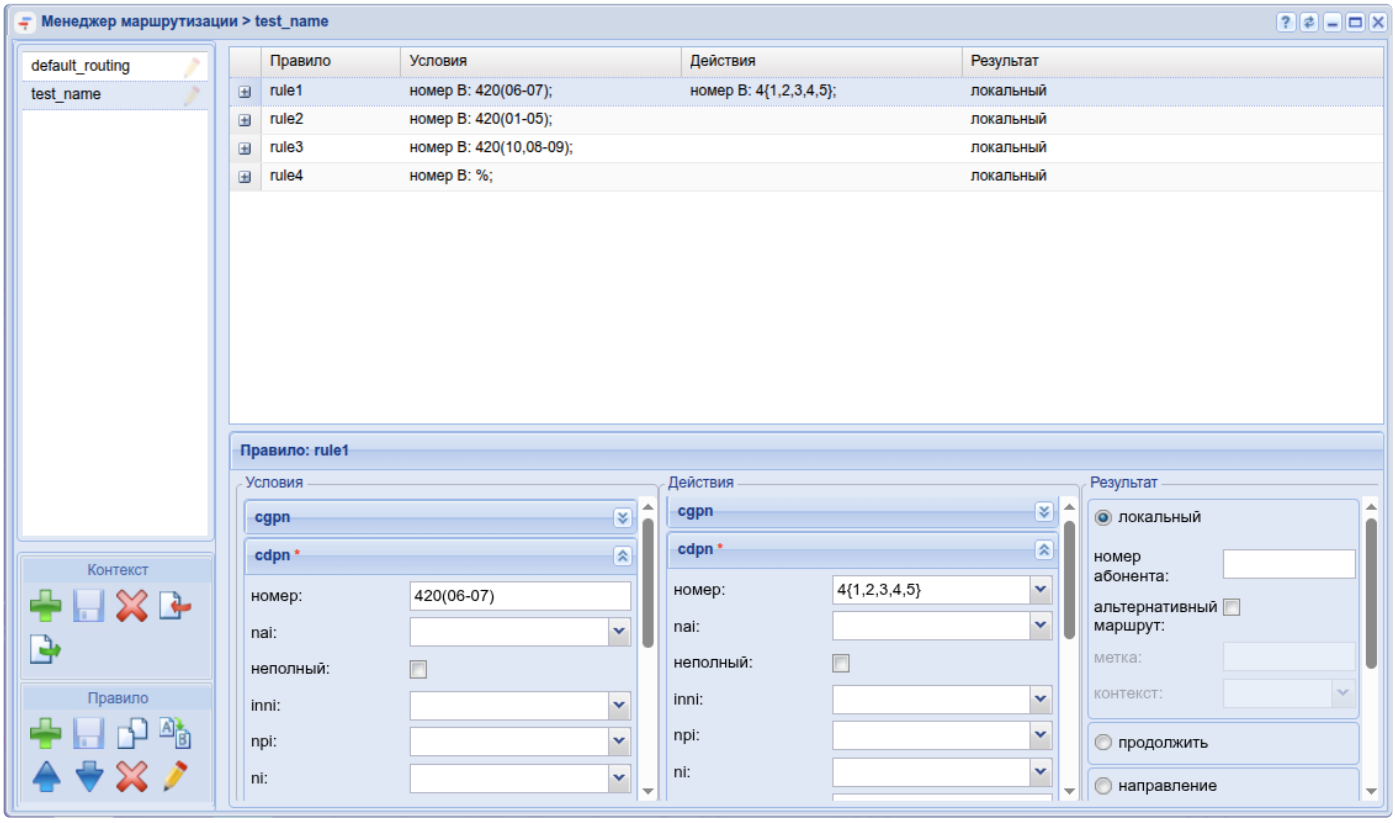


Рисунок 28 — Пример настроек контекста маршрутизации

9.9.9 Создание транка

Для создания и редактирования параметров транка в системе используется приложение "Менеджер транков" ("Trunk manager") .

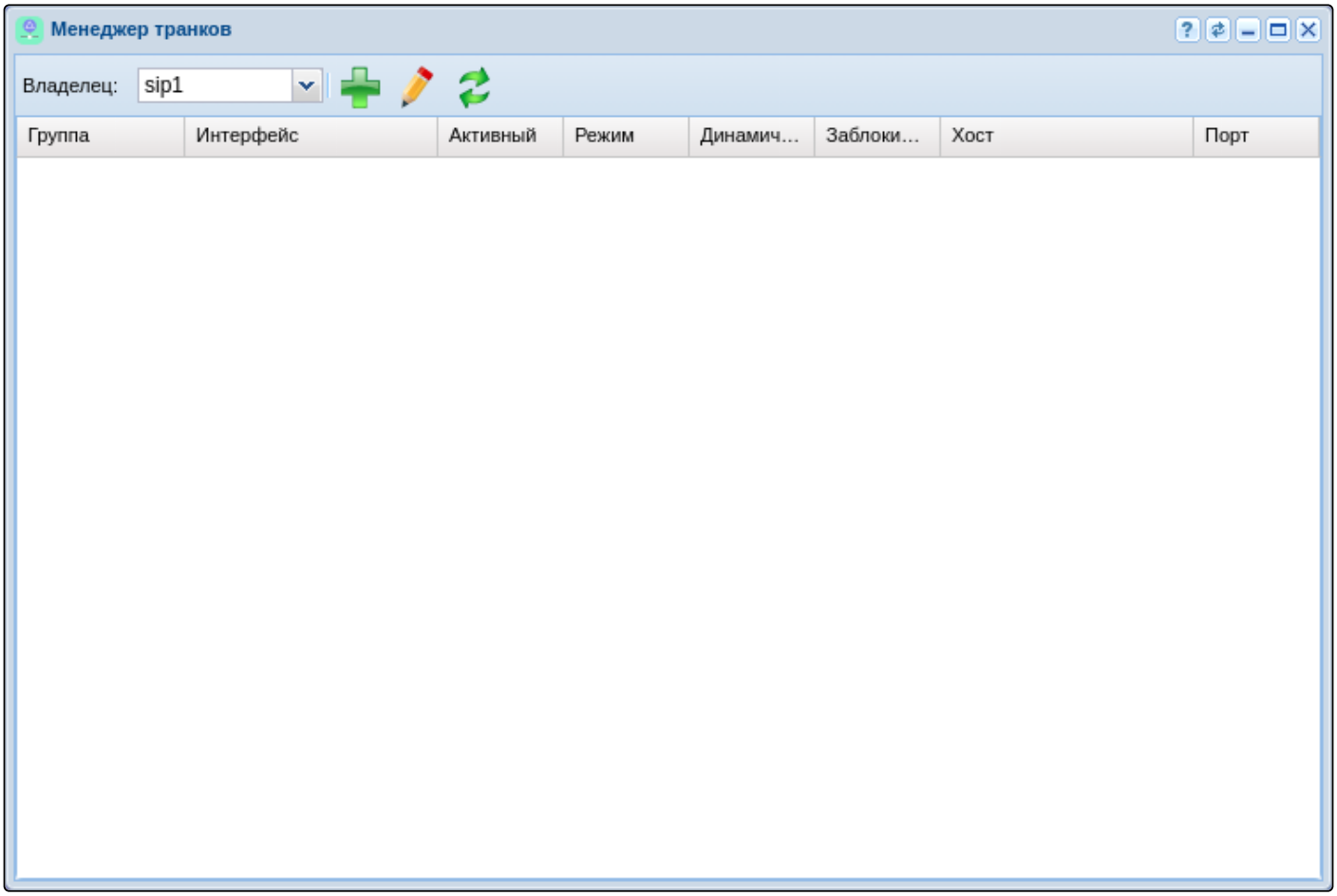


Рисунок 29 — Вид приложения "Менеджер транков"

Для определения транка в системе нужно нажать на кнопку *"Создать транк"*  , в появившемся диалоговом окне определить параметры:

- *Имя* — назначить название транка, по которому его можно идентифицировать в системе;
- *Контекст* — применить ранее созданный контекст маршрутизации test_name;
- *Группа* — выбрать группу интерфейсов, созданную при определении абонентов "test.group";
- *Группа IP-адресов (IP-set)* — созданная на домене группа адресов "test_set";
- *Регистрация* — если используется транковая, то нужно включить данный параметр, в примере она не используется;
- *Хост [:порт]* — IP-адрес назначения транка — 10.0.3.100;
- *Порт для приёма* — транспортный порт, на котором будет слушаться трафик с транка, соответствует порту, назначенному на группу IP-адресов.

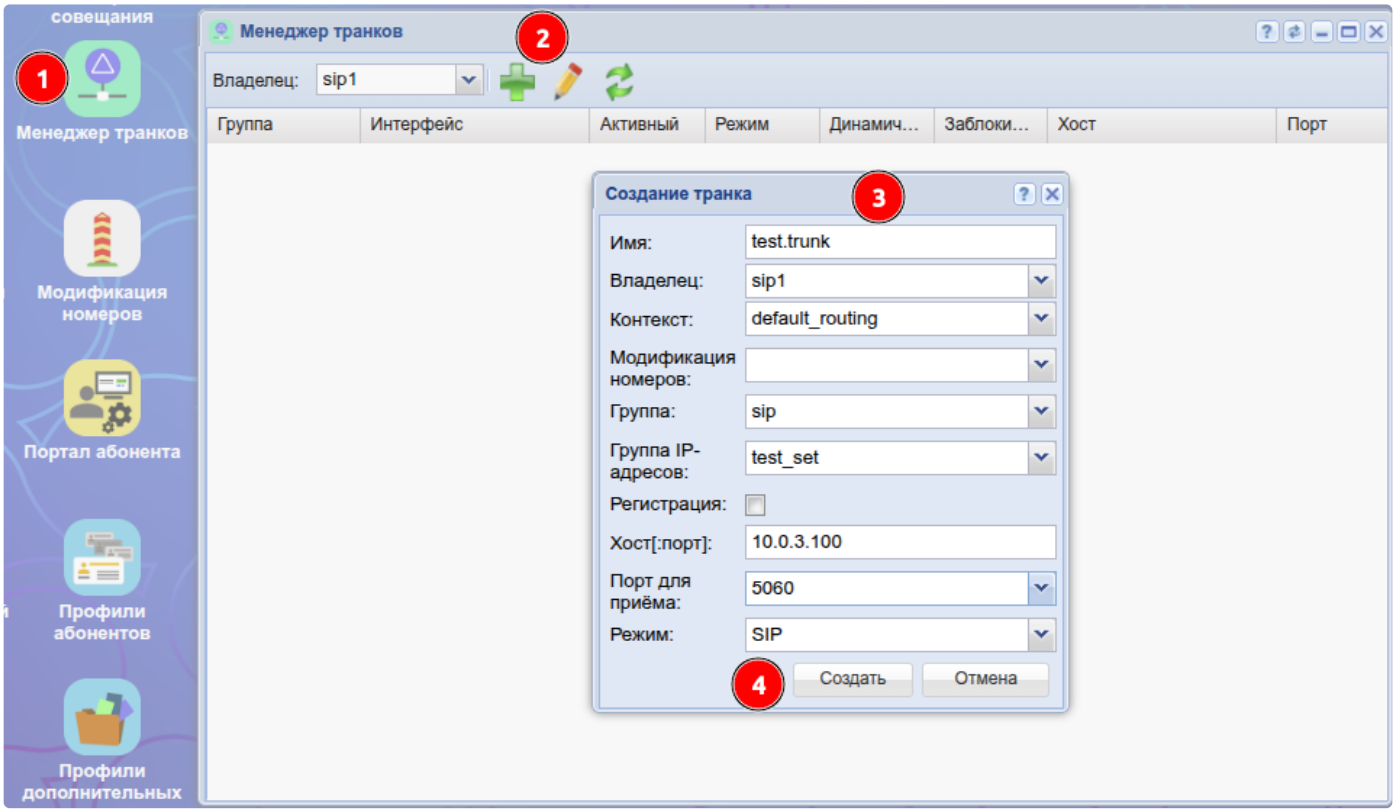


Рисунок 30 — Создание транка

9.9.10 Создание IVR-скрипта

Для создания IVR используется приложение "IVR-редактор" ("IVR editor").

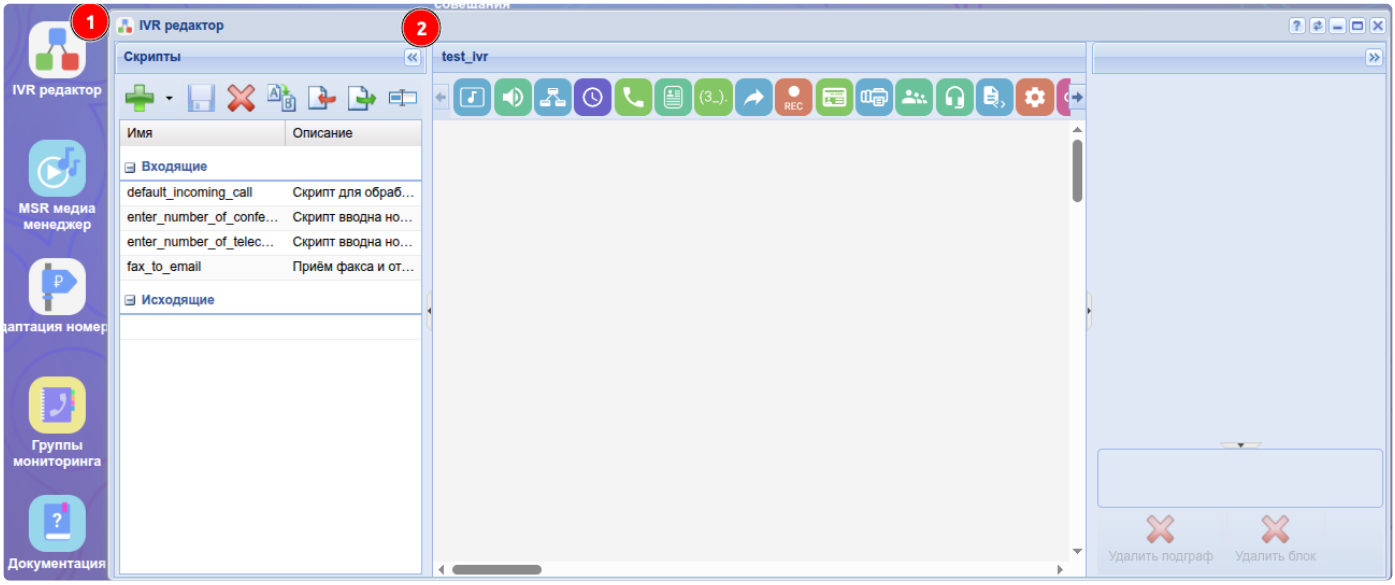


Рисунок 31 — Вид приложения "IVR-редактор"

Для создания скрипта нужно нажать на кнопку *"Добавить"* , выбрать тип скрипта (в данном случае будет скрипт на входящие вызовы), в диалоговом окне указать название скрипта, например "test_ivr".

После создания скрипта в основном окне редактора появится блок-схема.

В качестве примера составим скрипт, который при срабатывании проигрывает вызывающему абоненту заранее записанную фразу, а затем продолжает вызов.

Пример скрипта приведен на рисунках ниже. Здесь используются блоки Info и Dial.

- Блок Info — блок, который проигрывает сообщения до момента ответа пользователя. В качестве записанной фразы используется генератор тонов:

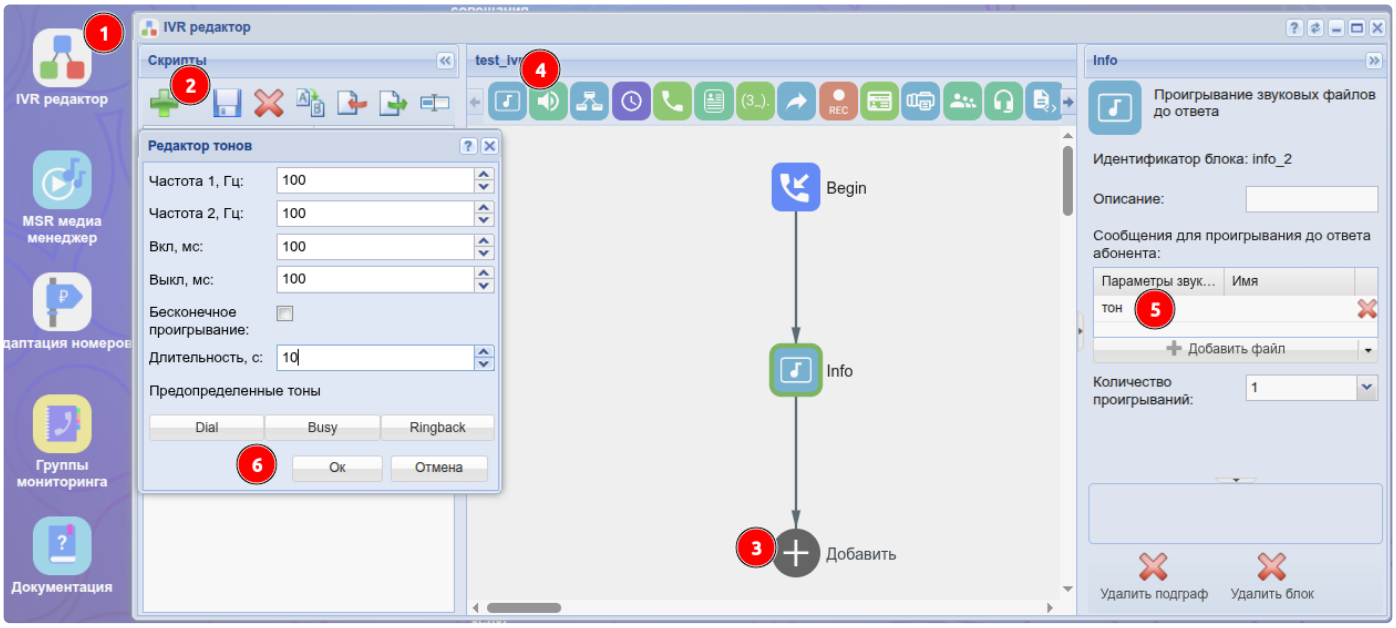


Рисунок 32 — Вид рабочей области IVR с настройками блока "Info"

- Блок dial — блок, который совершает вызов на заданный номер. Чтобы продолжить вызов, нужно применить предопределенную переменную CDPN:

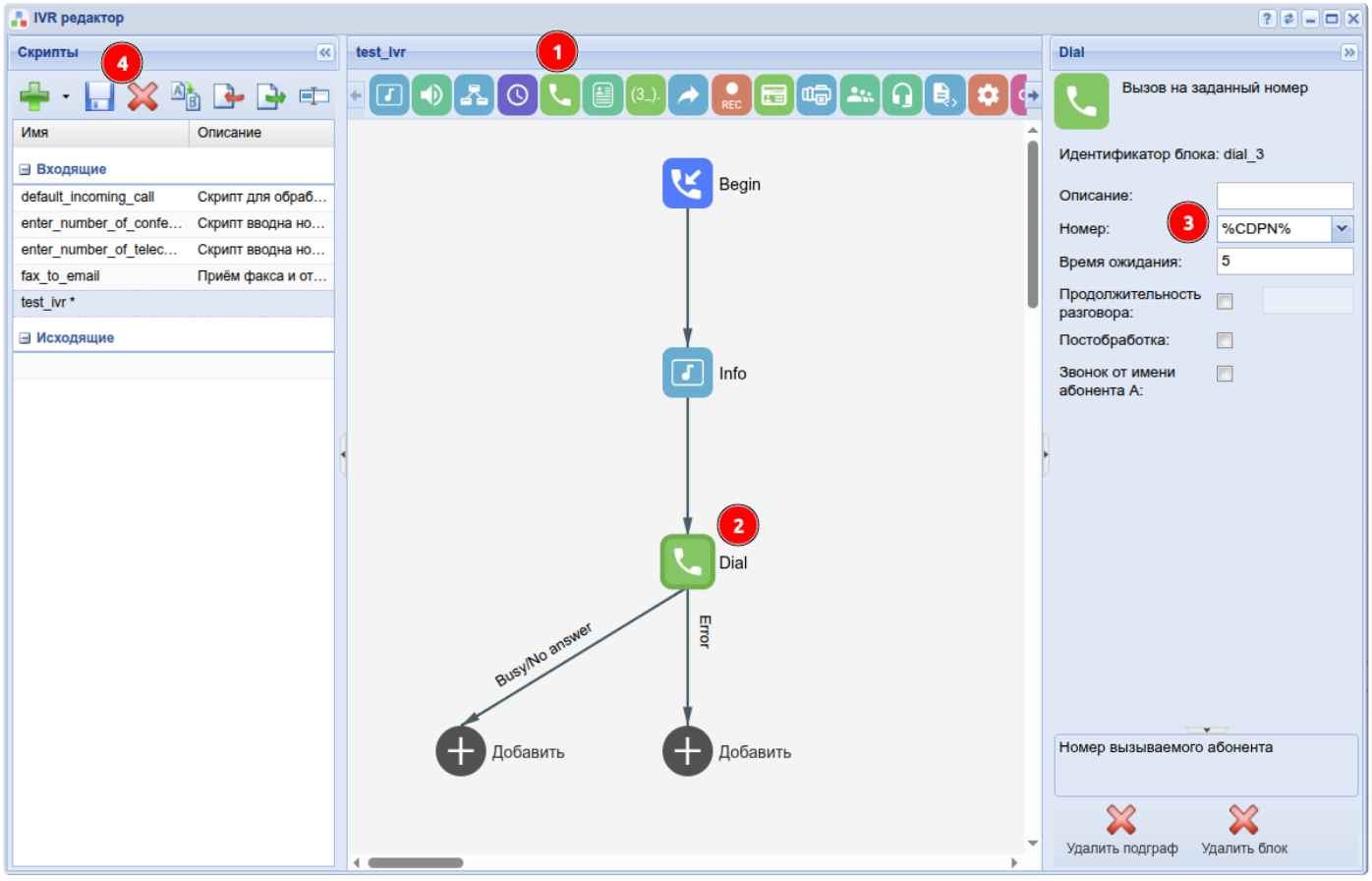


Рисунок 33 — Вид рабочей области IVR

9.9.11 Завершение настройки маршрутизации

Для завершения настройки маршрутизации нужно войти в приложение "Менеджер маршрутизации" ("Routing manager") и в правилах rule1 и rule3 настроить соответствующие результаты маршрутизации, для этого:

- rule1: в разделе "Результат" выберите подпункт "внешний". В таблице "Направления" в поле "Значение" добавьте созданный ранее транк, после чего сохраните правило .
- rule3: В разделе "Результат" выберите подпункт "ivr" и в поле скрипт добавьте созданный ранее ivr-скрипт, после чего сохраните правило и контекст .

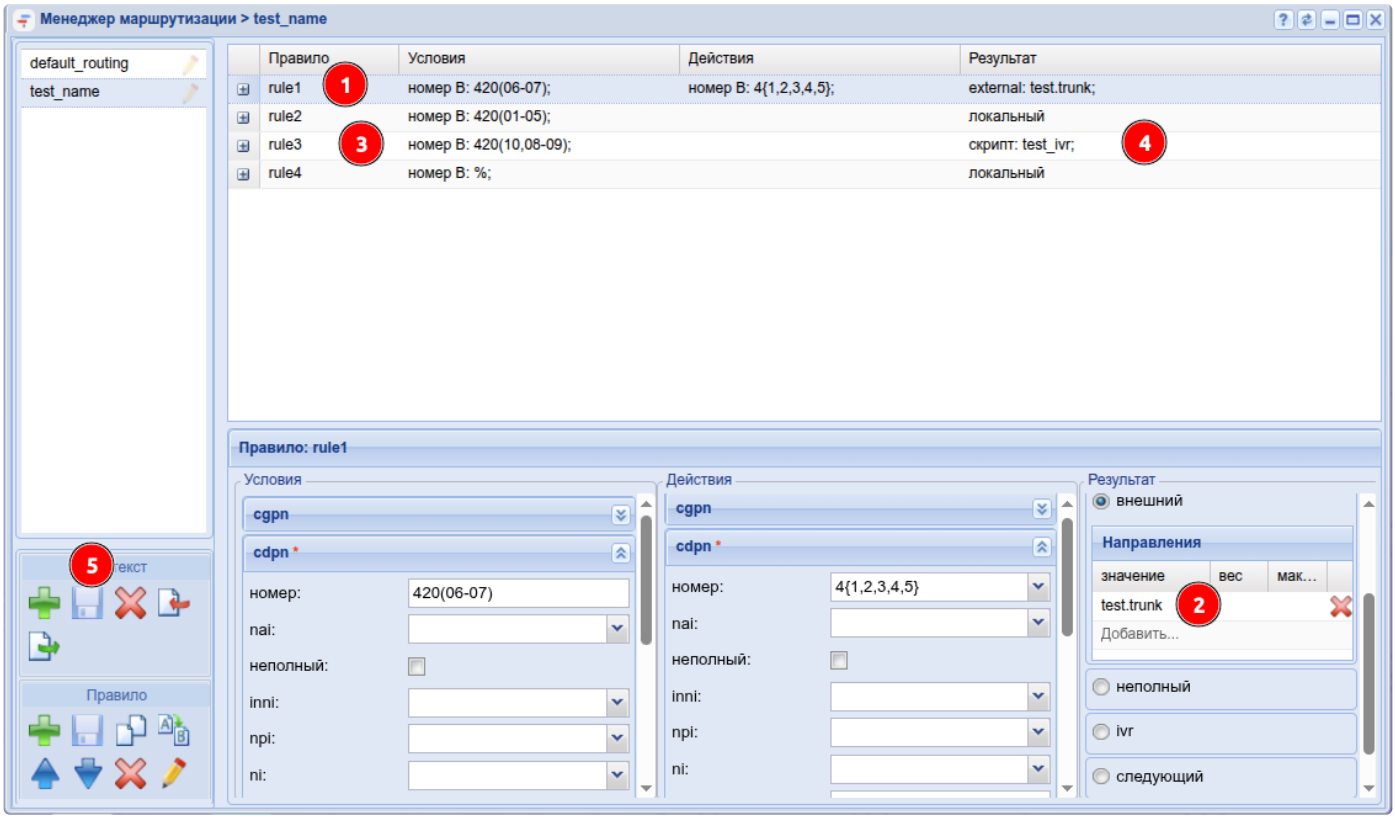


Рисунок 34 — Вид контекста маршрутизации

9.10 Первоначальная конфигурация системы через интерфейс командной строки (CLI)

пример первоначальной конфигурации через CoCon (CLI)

9.10.1 Добавление медиа-ресурсов в интерфейсе командной строки (CLI)

Перейдите в CLI CoCon:

на ecss1 пароль - password

```
ssh admin@localhost -p 8023
```

Медиаресурс — это описание параметров медиасервера, необходимых для работы с ним.

⚠ Для возможности использования медиаресурса и установления соединений , необходимо выполнить активизацию в команде создания медиаресурса - **/system/media/resource/declare** (параметр **"is active"=true**) или **/system/media/resource/set** (параметр **"active"=true**), если при создании активизация не была выполнена. Так же можно выполнить активизацию через веб интерфейс , приложение - сетевые окончания MSR (MSR-registrars).

declare

declare
Команда для объявления подключения медиасервера к ECSS-10.

Для работы в режиме предварительно настроенной системы можно заранее прописать медиаресурсы, информация о которых известна.

Для этого необходимо создать описание известных медиасерверов с указанием всех параметров, что позволить начать работу с медиасервером сразу после его регистрации.

Команда для создания конфигурации медиасервера:

Путь команды:

/system/media/resource/declare

Синтаксис:

declare <node> by_contact <contact> <zone> <site> <is active>

или

declare <node> by_iface <msr> <acc_id> <iface> <zone> <site> <is active>

Параметры:

<node> — имя ноды core;
<contact> — contact uri декларируемого медиаресурса;
<zone> — логическая зона обслуживания MSR;
<site> — идентификатор сети обслуживания MSR;
<is active> — состояние активности медиаресурса, значения true/false;
<msr> — имя MSR;
<acc_id> — номер идентификатор ресурса;
<iface> — название интерфейса.

set

set

Команда для изменения настроек медиаресурсов.

Путь команды:

/system/media/resource/set

Синтаксис:

set <node|*> <contact> <active|zone|site> <value>

Параметры:

<node|*> - имя ноды;
<contact> - contact uri декларируемого медиаресурса;
<active|zone|site> - активация | зона | сайт;
<value> - значение параметра.

Пример:

выполняется в CoCon

```
/system/media/resource/declare core1@ecss1 contact net.20@msr.ecss1 default local true
```

Результат:

```
/system/media/resource/declare core1@ecss1 contact net.20@msr.ecss1 default local
true
  Declared media resources
┌──────────────┬──────────────────────┬────────┬─────────┬────────┐
│     Node     │       Contact        │ Active │   Zone  │  Site  │
├──────────────┼──────────────────────┼────────┼─────────┼────────┤
│ core1@ecss1  │ net.20@msr.ecss1     │ true   │ default │ local  │
└──────────────┴──────────────────────┴────────┴─────────┴────────┘
```

info

info
Команда для просмотра информации о состоянии подключения конкретного медиаресурса.

Путь команды:

/system/media/resource/info

Синтаксис:

info <node> <contact>

Параметры:

<contact> — contact uri декларируемого медиаресурса;
<node> — имя ноды.

Пример:

```
/system/media/resource/info core1@ecss1 net.20@msr.ecss1
  Media resource specific for node core1@ecss1:
┌─┬──────────────────┬──────────────────────┐
│F│    Parameter     │        Value         │
├─┼──────────────────┼──────────────────────┤
│r│ Contact          │ net.20@msr.ecss1     │
│r│ Status           │ registered           │
│r│ Cc-status        │ connected            │
│r│ Cc-status uptime │ 00:16:48             │
│r│ Cc-id            │ cb6599               │
│r│ Address          │ 10.0.20.41:5700      │
│r│ Register time    │ 2025/12/03 09:40:25  │
│r│ Expired          │ 131                  │
│r│ Iface            │ net.20(10.0.20.41)   │
│ │ Active           │ true                 │
│ │ Zone             │ default              │
│ │ Site             │ local                │
│r│ MSR version      │ 3.18.1.0.12          │
│r│ MSR load         │ 0                    │
│r│ MSR perf coef    │ 1.0                  │
```

```
└─┴──────────────────┴──────────────────────────┘
Legend:
Column F - means Flag, values r - flag for readonly parameter
```

list

list
Команда для просмотра списка медиаресурсов в системе ECSS-10.

Путь команды:

/system/media/resource/list

Синтаксис:

list [active | all | register]

Параметры:

active — отображать только активные медиаресурсы (по умолчанию);
register — отображать только зарегистрированные медиаресурсы;
short — отобразить минимальную информацию;
all — отображать все задекларированные медиаресурсы.
По умолчанию команда выдает список зарегистрированных медиасерверов.

Пример:

```
/system/media/resource/list all
  All media resource selected list specific:
┌──────────────┬──────────┬──────────────┬──────────┬──────┬────────┬──────────┬──────────┬──────────┐
│    Node      │    MSR   │     MSR      │    MSR   │ MSR  │ Cc-id  │ Cc-status │ Cc-uptime │   Cc-address    │ Iface  │   Iface    │ Active │  Zone   │ Site  │ Contact      │   Status   │ Expired │
│              │          │   version    │ perf coef │ load │        │          │          │
│              │  name  │    addr    │          │       │
│            │         │
├──────────────┼──────────┼──────────────┼──────────┼──────┼────────┼──────────┼──────────┼──────────┤
│ core1@ecss1 │ msr.ecss1 │ 3.18.1.0.12  │       1.0 │    0 │ 54faa9 │ connected │ 00:54:48  │ 10.0.20.41:5700 │ net.20 │ 10.0.20.41 │ true   │ default │ local │ net.20@msr.ecss1 │ registered │     129 │
└──────────────┴──────────┴──────────────┴──────────┴──────┴────────┴──────────┴──────────┴──────────┘
```

9.10.2 Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI) CoCon

/system/media/registrar/ - команды управления регистратором для MSR медиа менеджера

set

set

Команда для изменения общих и индивидуальных параметров.

Путь команды:

/system/media/registrar/set

Синтаксис настройки общих параметров:

set common <FIELD> <VALUE>

Параметры:

<FIELD> - имя настраиваемого параметра;

<VALUE> - значение настраиваемого параметра

Где

- common - тип выставляемого параметра;
- <common parameter> - общий параметр медиарегистратора для всех нод: registration-timeout;
- private - тип выставляемого параметра;
- <node name> - имя ноды на которой находится медиарегистратор;
- <private parameter> - параметр медиарегистратора, который выставляется на <node name>: is-active, listen-interface & listen-port.

Список параметров доступных для настройки:

- is-active - позволяет включить или выключить регистратор (false - деактивирует регистратор, регистратор перестает принимать SIP трафик);
- listen-interface - имя интерфейса, который использует регистратор;
- listen-port - порт, на котором регистратор принимает запросы на регистрацию (по умолчанию 5000);
- registration-timeout - таймер перерегистрации, диапазон значений от 60 до 86400.

В общих параметрах доступна настройка времени регистрации, по умолчанию установлено 60 секунд.

Пример:
Установить время регистрации 60 секунд:

```
/system/media/registrar/set common registration-timeout 60
Common parameter registration-timeout set to 60.
```

Синтаксис настройки индивидуальных параметров:

system/media/registrar/set private <CORE> <FIELD> <VALUE>

Параметры:

<CORE> - нода ядра, для которой настраивается Registrar-сервер;

<FIELD> - имя настраиваемого параметра;

<VALUE> - значение настраиваемого параметра.

Настраиваемые параметры:

- is-active - активен ли Registrar-сервер, принимает значения true | false;
- listen-ip - ip-адрес, на котором Registrar-сервер будет принимать соединения (по умолчанию 0.0.0.0);
- listen-port - сетевой порт, на котором Registrar-сервер будет принимать соединения (по умолчанию 5000).

Пример:

Установить ipадрес для приема соединений от медиасервера:

выполняется в CoCon

```
/system/media/registrar/set private core1@ecss1 listen-ip 10.0.20.41
```

Результат:

```
/system/media/registrar/set private core1@ecss1 listen-ip 10.0.20.41
Private parameter listen-ip set to 10.0.20.41 at node core1@ecss1.
```

info

info

Команда просмотра текущих настроек Registrar-сервера.

Путь команды:

/system/media/registrar/info

Пример:

```
/system/media/registrar/info
 Common parameters:
┌──────────────────────────┬─────────┐
│         Parameter        │ Value   │
├──────────────────────────┼─────────┤
│ registration-timeout, sec│   180   │
└──────────────────────────┴─────────┘
 Private parameters:
┌──────────────┬──────────┬────────────┬──────────────┬──────────────┐
│    Node      │ Status   │ Is-active  │ Listen-ip    │ Listen-port  │
├──────────────┼──────────┼────────────┼──────────────┼──────────────┤
│ core1@ecss1  │ active   │ true       │ 10.0.20.41   │        5000  │
└──────────────┴──────────┴────────────┴──────────────┴──────────────┘
```

Также есть возможность добавить медиа-ресурсы через приложение web-конфигуратора — сетевые окончания MSR (MSR-registrars).

После того как все сервисы будут запущены, ноды какое-то время будут устанавливать связь. Как только все ноды загрузятся , при выполнении запроса **system-status**, последует вывод следующей информации:

```
/system-status
Checking...
┌─┬──────────────┬──────────────────────────┬──────────────┬──────────────┐
└─┘
│ │     Node     │          Release         │  Erlang nodes  | Mnesia nodes
```

```
|Uptime|
├─┼──────────────┼──────────────────────────────┼──────────────┼──────────────┼──────
──┤
| |core1@ecss1    |ecss-core-3.18.1.0.1657     |core1@ecss1     |not running   |1h
23m|
| |ds1@ecss1      |ecss-ds-3.18.1.0.1657       |ds1@ecss1       |ds1@ecss1     |1h
23m|
| |md1@ecss1      |ecss-mediator-3.18.1.0.1657 |md1@ecss1       |md1@ecss1     |1h
23m|
| |mycelium1@ecss1|ecss-mycelium-3.18.1.0.1657 |mycelium1@ecss1 |not running   |1h
23m|
| |sip1@ecss1     |ecss-pa-sip-3.18.1.0.1657   |sip1@ecss1      |sip1@ecss1    |1h
23m|
└─┴──────────────┴──────────────────────────────┴──────────────┴──────────────┴──────
──┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────────┬────────────┬──────────────┐
│     Node     │    MSR   │     MSR      │ Cc-status  │ Cc-uptime    │
│              │          │   version    │            │              │
├──────────────┼──────────┼──────────────┼────────────┼──────────────┤
│ core1@ecss1  │ msr.ecss1 │ 3.18.1.0.12 │ connected  │ 00:04:15     │
└──────────────┴──────────┴──────────────┴────────────┴──────────────┘
```

В примере выше видно, что ноды стартовали и *MSR* зарегистрировался на ноде *ecss-core*.

9.10.3 Создание и настройка группы SIP-IP-адресов (IP-set)

Произведите настройку адаптера SIP согласно ТЗ:

выполняется в CoCon

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss1 ip = 10.0.20
.41
```

выполняется в CoCon

```
/cluster/adapter/sip1/sip/network/set ip_set test_set listen-ports list = [5060]
```

Результат:

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss1 ip
= 10.0.20.41

Property "ip_set" successfully changed from:

        to
test_set: no ports set
```

```
test_set: sip1@ecss1 10.0.20.41
test_set: dscp 0.

/cluster/adapter/sip1/sip/network/set ip_set test_set listen-ports list = [5060]
Property "ip_set" successfully changed from:
test_set: no ports set
test_set: sip1@ecss1 10.0.20.41
test_set: dscp 0
        to
test_set: 5060
test_set: sip1@ecss1 10.0.20.41
test_set: dscp 0
```

9.10.4 Создание Домена и подключение IP-set

Создайте Домен (виртуальную АТС) и присвойте ему созданную выше группу (IP-set) настроек SIP-адаптера:

выполняется в CoCon

```
/domain/declare test_domain --add-domain-admin-privileges --add-domain-user-privileges
```

выполняется в CoCon

```
/domain/test_domain/sip/network/set ip_set [test_set]
```

Результат:

```
/domain/declare test_domain --add-domain-admin-privileges --add-domain-user-
privileges
New domain test_domain is declared

/domain/test_domain/sip/network/set ip_set [test_set]
Property "ip_set" successfully changed from:
[]
 to
["test_set"].
```

После создания домена, администратор системы должен открыть доступ на управление услугами оператору Домена (виртуальной АТС). Для этого использовать команду:

Разрешения оператору Домена (виртуальной АТС) "test_domain" управлять всеми услугами :

выполняется в CoCon

```
/cluster/storage/ds1/ss/access-list add test_domain *
```

Результат:

```
/cluster/storage/ds1/ss/access-list add test_domain *
Supplementary services successfully added in the domain test_domain.
```

Для возможности применения пакета услуг у абонентов сразу после создания, необходимо настроить политику применения лицензионных пакетов услуг (детальную информацию вы можете посмотреть здесь)

Создать политику применения лицензии с именем -"new_sub"

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/declare new_sub "Задаются при создании абонента"
```

Результат:

```
/domain/test_domain/ss/licence/politics/declare new_sub "Задаются при создании
абонента"
Licence politic new_sub successfully declared.
```

Задать имя пакета услуг дополнительных сервисов, для этого использовать следующие команды :

выполняется в CoCon user/pw admin/password

```
/cluster/storage/ds1/licence/show-licence 1
```

нас интересует следующая информация :

```
/cluster/storage/ds1/licence/show-licence 1
SSW ID                                                  < Имя SSW ID >
Description                                             ECSS ELM License

. . .

Supplementary Services licence package(s)
 name                                                   'ECSS-ADV'
 limit                                                  600000
 description                                            "Весь пакет услуг"
 SS list                                                [1,2,3,4,5]

. . .
```

имя пакета услуг в примере "**ECSS-ADV**".
Добавить в созданную политику пакет лицензионных услуг с именем "ECSS-FULL+", именно так он описан в лицензии:

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/package-add new_sub ECSS-ADV
```

Результат:

```
/domain/test_domain/ss/licence/politics/package-add new_sub ECSS-ADV
Licence packages ["ECSS-ADV"] successfully added to politic new_sub.
```

Пакет добавлен, но в настоящий момент политика не активна (не может быть применена к абоненту). Для активации политики new_sub выполнить следующую команду:

выполняется в CoCon user/pw admin/password

```
domain/test_domain/ss/licence/politics/activate new_sub
```

Результат:

```
/domain/test_domain/ss/licence/politics/activate new_sub
Licence politic new_sub successfully activated.
```

Теперь при создании абонентов можем использовать дополнительные услуги.

10 Установка ECSS10 3.18 на Astra Linux, в открытом контуре, установка в кластере

10.1 Первоначальная установка системы с резервированием, в кластере из двух серверов

10.1.1 Исходные данные

Вариант с резервом, используется на проектах чаще , поэтому его мы рассмотрим первым. Пример установки на одном сервере

Конфигурация системы в кластере из двух серверов

Исходные данные

В данном разделе использованы специфичные для данного продукта термины, ознакомиться с ними можно здесь.
Техническое задание:
Требуется интеграция программного коммутатора 5 класса ECSS-10 (SSW) на 2-х физических серверах с поддержкой SIP со следующими параметрами на нагрузку группы серверов:

- Максимальное количество абонентов 15000 (MUL — Max user limit);
- Максимальное количество одновременных соединений 2000 (MCL — Max call limit);
- Требуется полный резерв системы ecss (все node резервируется на втором сервере);
- Количество сетевых интерфейсов ethernet — 3/5.

По техническому заданию требуется определить аппаратную платформу.

Полоса пропускания СПД

Требуемая полоса пропускания СПД **не менее** 1000Mb/s

Пример составления аппаратных требований:

Пример составления аппаратных требований:

Пример разнесения составляющих по адресному пространству

Имя сервера (хоста)	Роль	Интерфейс	Адрес/Маска	Порт
Статические адреса программного коммутатора				
ecss1	Интерфейс управления сервером (порт 2000 ssh)	eth0	192.168.56.41/24	2000
ecss2	Интерфейс управления сервером (порт 2000 ssh)	eth0	192.168.56.42/24	2000

ecss1	Адрес ядра (ecss-core)	net.20	10.0.20.41/24	5000
ecss2	Адрес ядра (ecss-core)	net.20	10.0.20.42/24	5000
ecss1	Адрес OAM	net.10	10.0.10.41/24	-
ecss2	Адрес OAM	net.10	10.0.10.42/24	-
ecss1, ecss2	Адрес шлюза	eth0	192.168.56.1	-
ecss1, ecss2	Адреса DNS-серверов	eth0	8.8.8.8	-
ecss1, ecss2	Адреса NTP-серверов	eth0	10.136.16.211, 10.136.16.212	123
Виртуальные адреса программного коммутатора				
ecss1	Виртуальный адрес программного адаптера хоста ecss1	net.20:SIP1	10.0.20.31/24	-
ecss2	Виртуальный адрес программного адаптера хоста ecss2	net.20:SIP2	10.0.20.32/24	-
ecss1	Запасной виртуальный адрес программного адаптера хоста ecss2 на хосте ecss1	net.20:SIP2	10.0.20.32/24	-
ecss2	Запасной виртуальный адрес программного адаптера хоста ecss1 на хосте ecss2	net.20:SIP1	10.0.20.31/24	-
Заполнить данные параметры для SSW из проектного технического задания				
Имя сервера (хоста)	**Роль**	**Интерфейс**	**Адрес/Маска**	**Порт**
Статические адреса программного коммутатора				
ecss1	Интерфейс управления сервером (порт 2000 ssh)			2000
ecss2	Интерфейс управления сервером (порт 2000 ssh)			2000
ecss1	Адрес ядра (ecss-core)			5000

ecss2	Адрес ядра (ecss-core)			5000
ecss1	Адрес OAM			-
ecss2	Адрес OAM			-
ecss1, ecss2	Адрес шлюза			-
ecss1, ecss2	Адреса DNS-серверов			-
ecss1, ecss2	Адреса NTP-серверов			123
Виртуальные адреса программного коммутатора				
ecss1	Виртуальный адрес программного адаптера хоста ecss1			-
ecss2	Виртуальный адрес программного адаптера хоста ecss2			-
ecss1	Запасной виртуальный адрес программного адаптера хоста ecss2 на хосте ecss1			-
ecss2	Запасной виртуальный адрес программного адаптера хоста ecss1 на хосте ecss2			-

Включение в сеть

Топологию включения сервера в сеть для обеспечения резервирования рекомендуется делать с использованием 2-х коммутаторов. Пример на рисунке ниже.

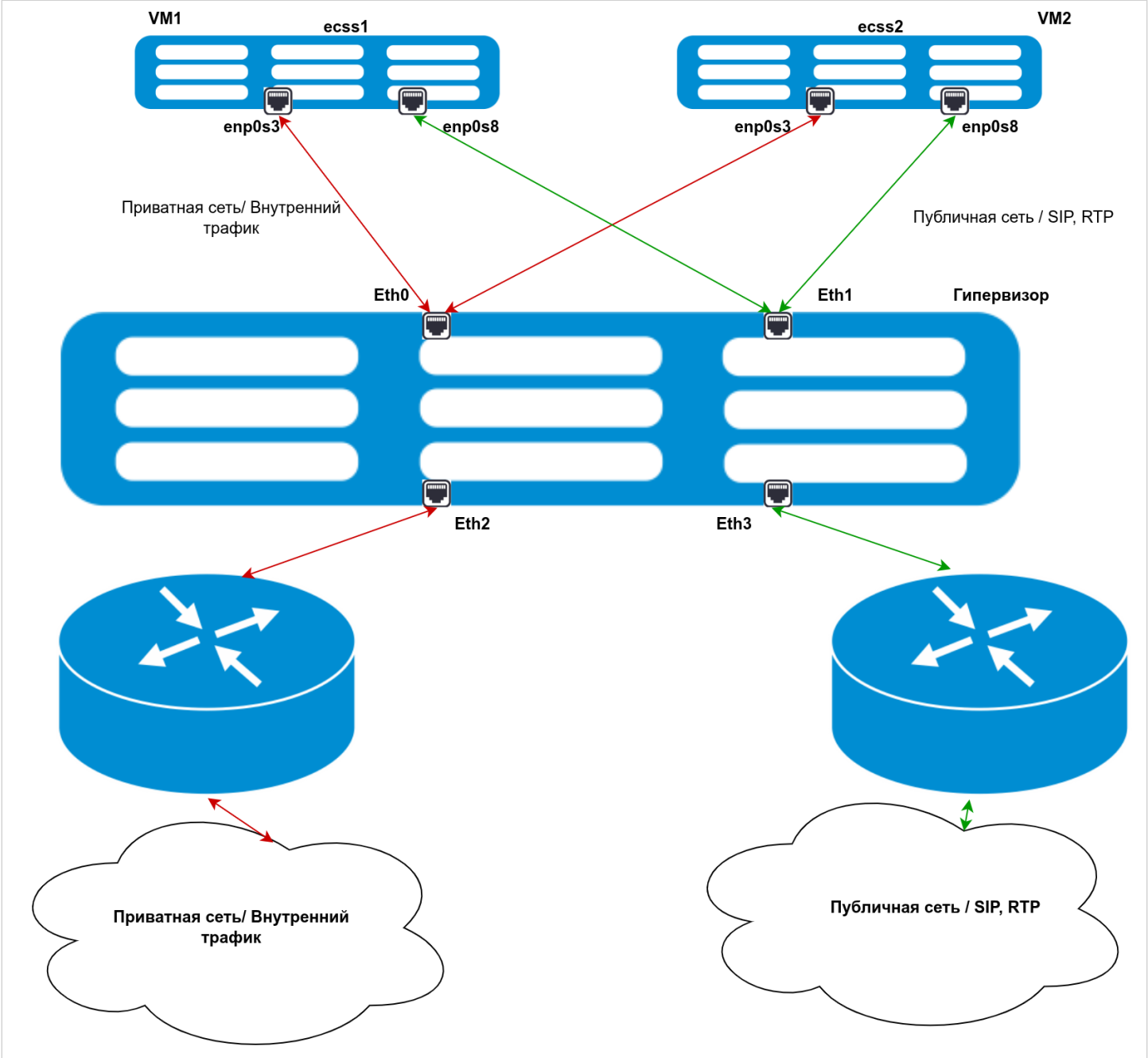


Рисунок 1 — Схема организации связи

10.1.2 Последовательность установки SSW ECSS10 в кластере из двух серверов

Установка SSW ECSS10 состоит из 2-х основных частей :

- Подготовительной - Установка ОС Astra Linux 1.8, обновлений пакетов ОС Astra Linux 1.8 , "оптимизация" ОС, установка дополнительных пакетов программ, подготовка сетевых интерфейсов, /etc/hosts, ssh-keygen
- Установка приложений Eltex ECSS10

Подготовительный этап

В этом разделе приведено описание инсталляции операционной системы, а также необходимых и дополнительных пакетов. Система ECSS-10 версии 3.14 работает под управлением ОС Astra Linux 1.8.

Предварительные требования

- Установочный загрузочный носитель с дистрибутивом операционной системы;
- Подготовленный сервер с обновленным BIOS, ILO (если есть), подключенная сеть для доступа в сеть Интернет;
- Выставленный первый приоритет загрузки с установочного носителя — USB Flash или CD/DVD в BIOS;
- Достаточный объем дискового пространства и памяти в соответствии с проектом.

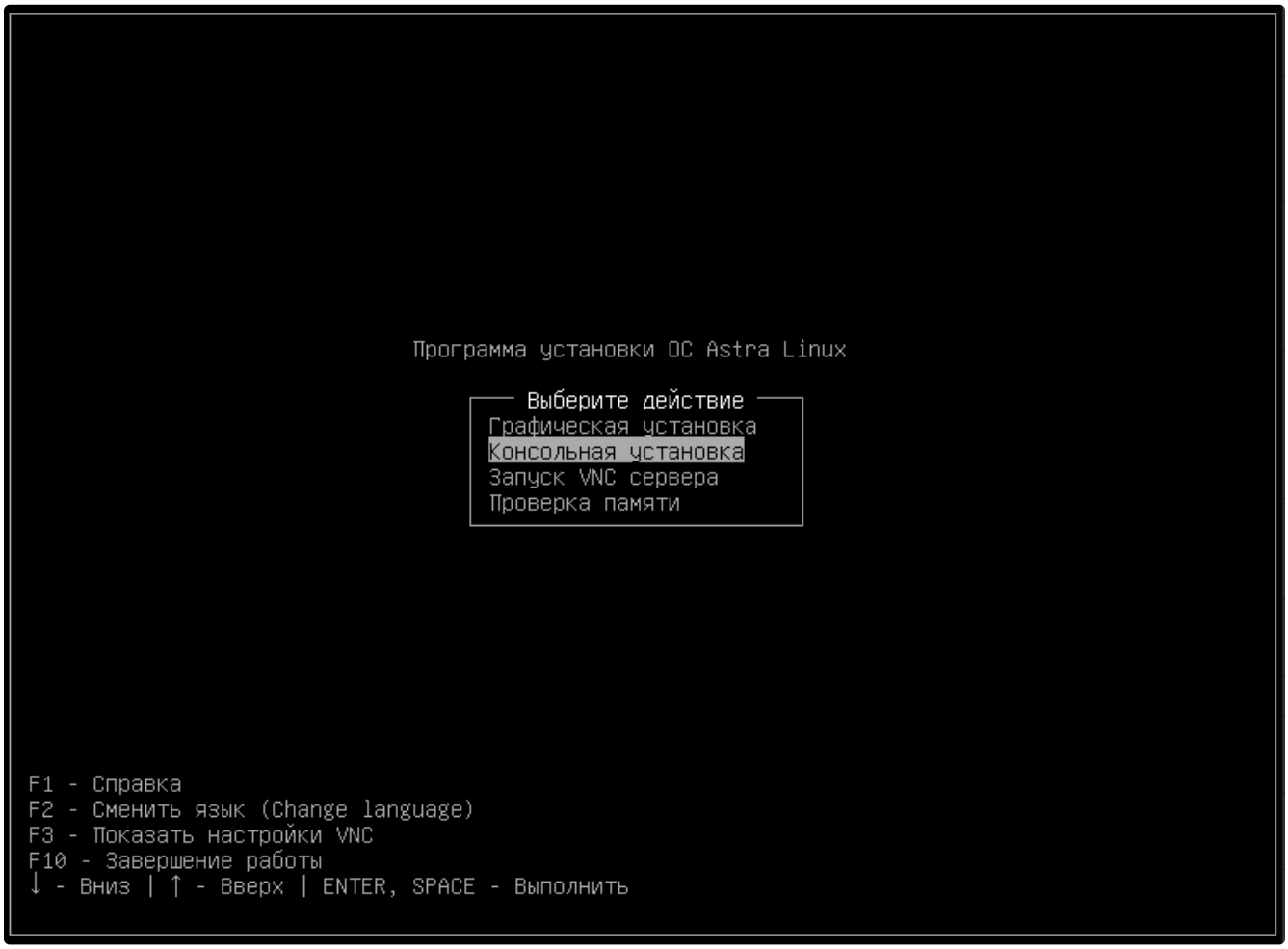
10.1.3 Установка ОС

Для установки ОС необходимо выполнить следующее:

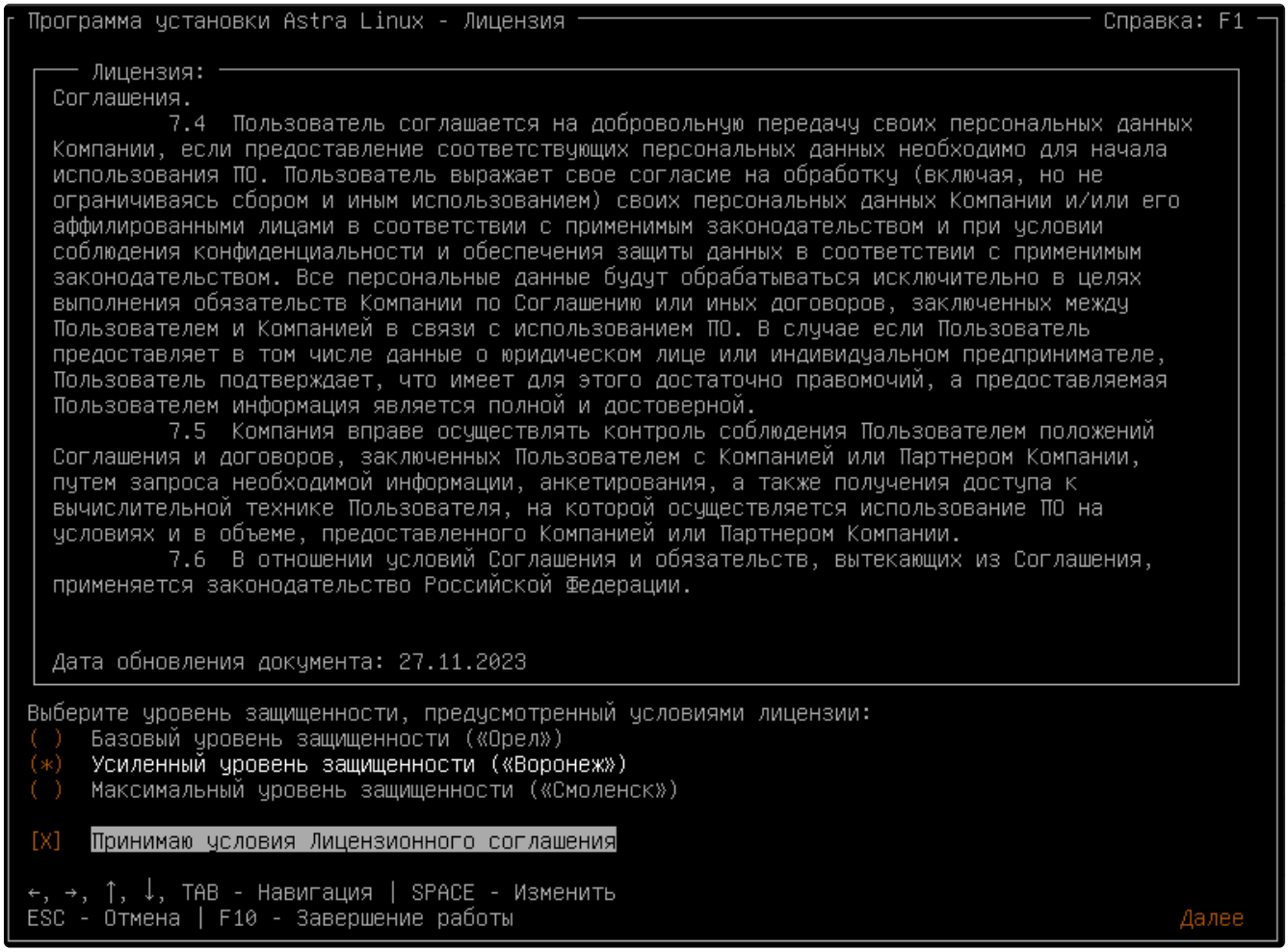
- После загрузки с установочного носителя выбрать "Графическая установка" или "Установка".
- Согласиться с лицензионным соглашением.
- Выбрать раскладку клавиатуры.
- Выбрать сетевой интерфейс для подключения к сети Интернет.
- Установить имя компьютера (рекомендованы ecss1/ecss2)
- Настроить пользователя- администратора.
- Создать партиции на диске в соответствии с таблицей 1 (разметка диска - Вручную).
- В процессе установке , **не создавать раздел swap**. При напоминании о пропуске раздела , **подтвердить** , что установка выполняется без создания swap.
- Установить часовой пояс.
- Установить OpenSSH server (активизировать пункт : "Средства удаленного подключения SSH").
- Выбрать уровень защищенности, который используется в вашем проекте
- Дополнительные настройки ОС - по умолчанию
- Перезагрузить успешно установленную ОС

Установка ОС Astra Linux 1.8

Выбрать режим установки



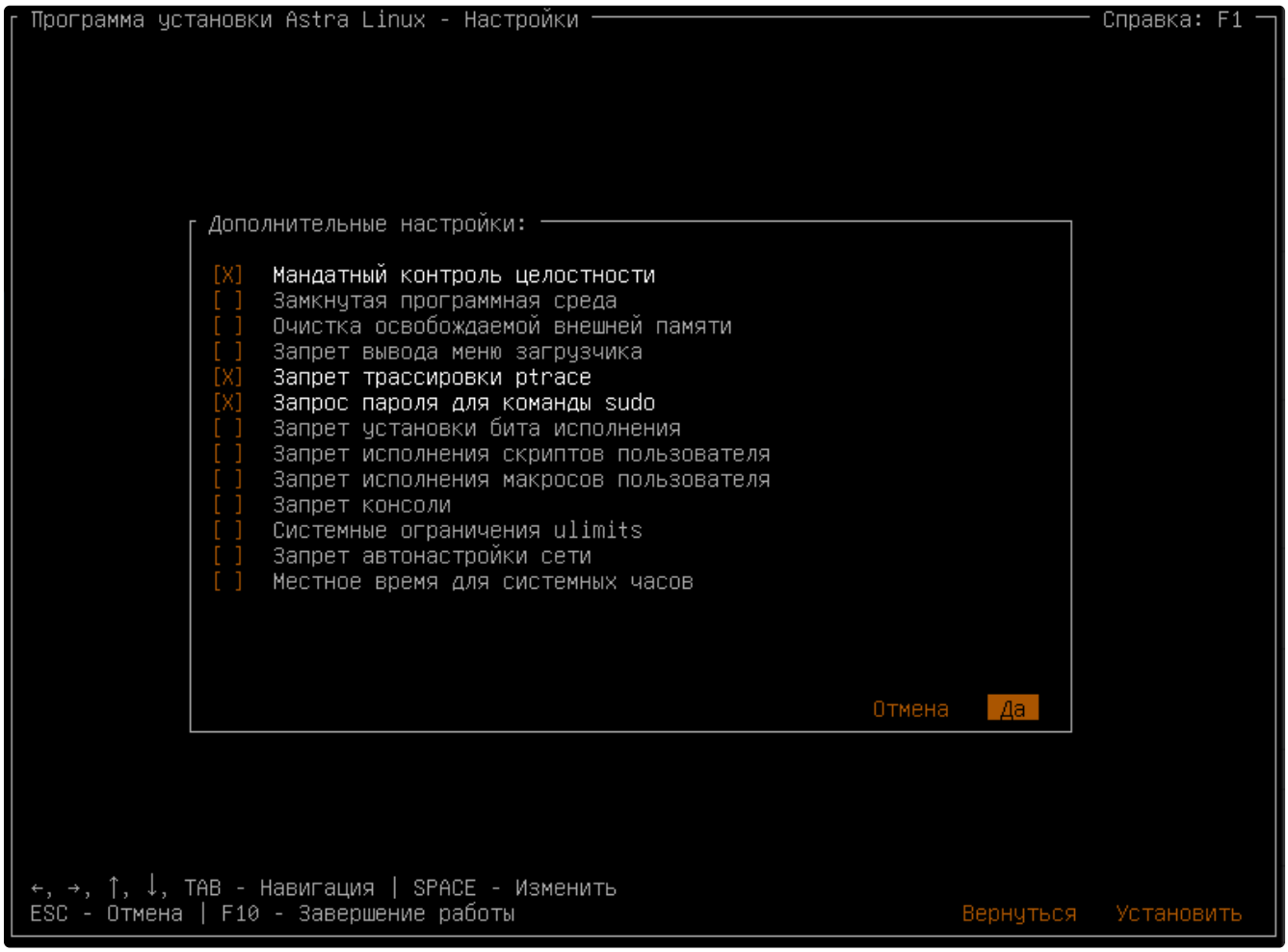
Согласится с Лицензионным Соглашением


Программа установки Astra Linux - Лицензия
Справка: F1
Лицензия:
Соглашения.
7.4 Пользователь соглашается на добровольную передачу своих персональных данных Компании, если предоставление соответствующих персональных данных необходимо для начала использования ПО. Пользователь выражает свое согласие на обработку (включая, но не ограничиваясь сбором и иным использованием) своих персональных данных Компании и/или его аффилированными лицами в соответствии с применимым законодательством и при условии соблюдения конфиденциальности и обеспечения защиты данных в соответствии с применимым законодательством. Все персональные данные будут обрабатываться исключительно в целях выполнения обязательств Компании по Соглашению или иных договоров, заключенных между Пользователем и Компанией в связи с использованием ПО. В случае если Пользователь предоставляет в том числе данные о юридическом лице или индивидуальном предпринимателе, Пользователь подтверждает, что имеет для этого достаточно правомочий, а предоставляемая Пользователем информация является полной и достоверной.
7.5 Компания вправе осуществлять контроль соблюдения Пользователем положений Соглашения и договоров, заключенных Пользователем с Компанией или Партнером Компании, путем запроса необходимой информации, анкетирования, а также получения доступа к вычислительной технике Пользователя, на которой осуществляется использование ПО на условиях и в объеме, предоставленного Компанией или Партнером Компании.
7.6 В отношении условий Соглашения и обязательств, вытекающих из Соглашения, применяется законодательство Российской Федерации.
Дата обновления документа: 27.11.2023
Выберите уровень защищенности, предусмотренный условиями лицензии:
( ) Базовый уровень защищенности («Орел»)
(*) Усиленный уровень защищенности («Воронеж»)
( ) Максимальный уровень защищенности («Смоленск»)
[X] Принимаю условия Лицензионного соглашения
←, →, ↑, ↓, TAB - Навигация | SPACE - Изменить
ESC - Отмена | F10 - Завершение работы
Далее

Настройки



Настойки (продолжение1)



Настройки (продолжение2)


Программа установки Astra Linux - Настройки
Справка: F1
Региональные настройки
Переключения языка: Ctrl+Shift
Системный язык: Русский
Часовой пояс: UTC+07:00 Россия (Новосибирск)
Дополнительные настройки: 27.11.2025, 09:12, NTP service d...
Настройки авторизации
Имя компьютера: astra-ecss
Имя администратора: eltex
Пароль: ***
Подтверждение пароля: ***
Настройки пароля GRUB: Совпадает с администраторским
Другие настройки
Компоненты ОС: Средства фильтрации сетевых паке...
Дополнительные настройки: Мандатный контроль целостности, ...
Дополнительные пакеты: Не выбраны
Ядро для установки: linux-6.1-generic (Рекомендуется)
Разметка диска: Использовать все пространство на...
[X] Перезагрузить компьютер после завершения установки
←, →, ↑, ↓, TAB - Навигация | SPACE - Изменить
ESC - Отмена | F10 - Завершение работы
Вернуться
Установить

Установка

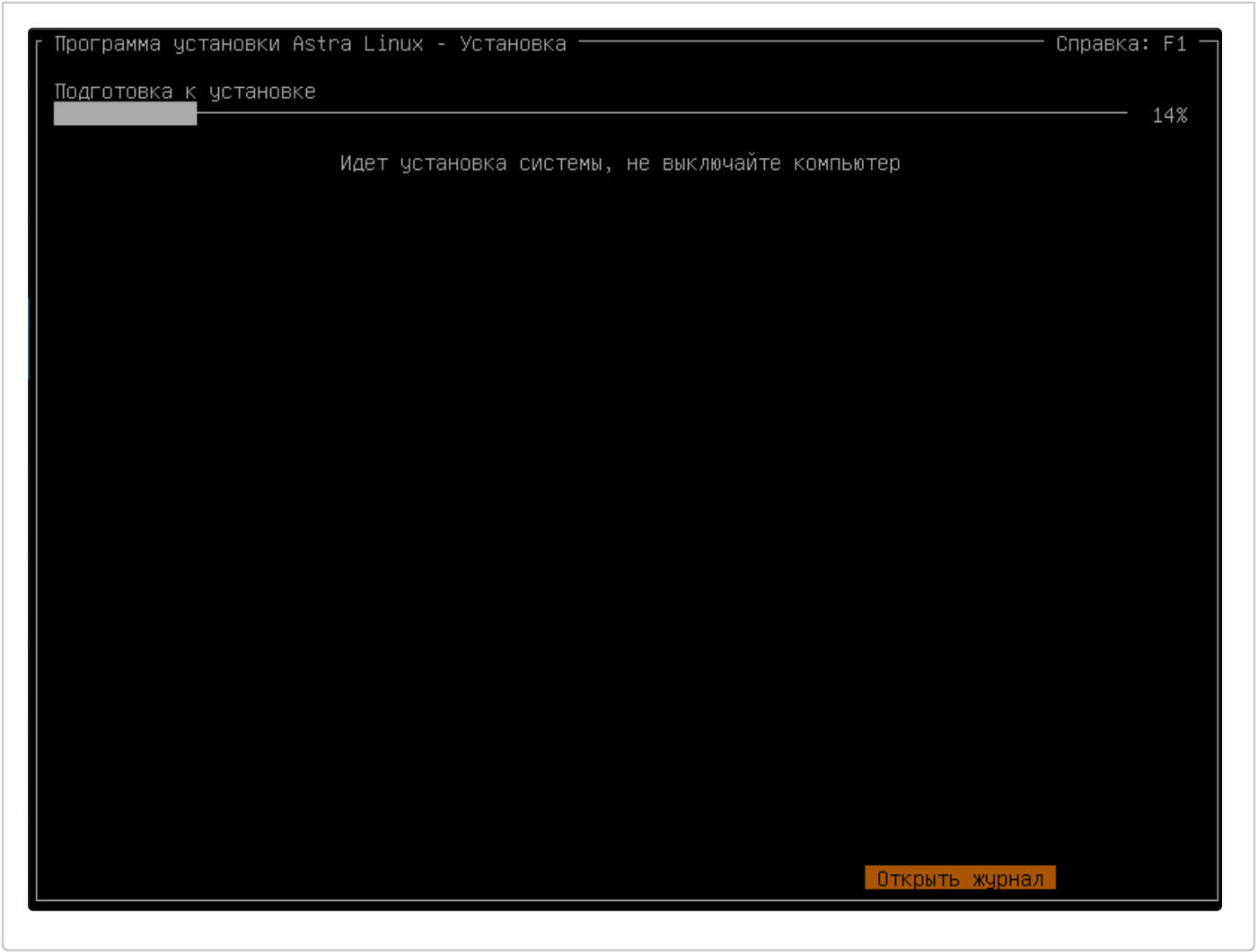


Таблица 1 — Вариант размещения информации в файловой системе на физических носителях для серверов

1	Загрузочный раздел операционной системы (создается автоматически)	boot	raid 1:hdd1,hdd2	boot	/boot	ext4	1 Gb	Первичный
2	Корневой раздел операционной системы	root	raid 1:hdd1,hdd2	root	/	ext4	30 Gb	Логический
3	Информация локальных баз данных	mnesia	raid 1:hdd1, hdd2	mnesia	/var/lib/ecss	ext4	30 Gb	Логический
4	Распределенная БД для хранения медиаресурсов	glusterfs	raid 1:hdd1, hdd2 или hdd3	glusterfs	/var/lib/ecss/glusterfs*	ext4	Max Gb	Логический

5	Журналы функционирования подсистем ОС	log	raid 1:hdd1,hdd2 или hdd3	log	/var/log	ext4	20 Gb	Логический
6	Журналы функционирования подсистем ECSS	ecss_log	raid 1:hdd1,hdd2 или hdd3	ecss_log	/var/log/ecss	ext4	20 Gb	Логический
7	Базы данных	ecss_db	raid 1:hdd1,hdd2 или hdd3	ecss_db	/srv/ecss/ecss-postgres-bdr-ssw/	ext4	100–400 Gb**	Логический
8	Файлы пользователя	home	raid 1:hdd1,hdd2 или hdd3	home	/home	ext4	10 Gb	Логический

⚠ * Если сервер не будет работать в кластере, то вместо **glusterfs** создается раздел **/var/lib/ecss/restfs**.
** Рекомендуемое значение для серий **Light, Light+, Midi** — 100 Gb. Рекомендуемое значение для серии **Heavy** — 200 Gb, **Super Heavy** — 400 Gb.

Для работы системы необходимо как **минимум** 256 Gb свободного пространства.

На серверах системы необходимо настроить параметр **"hostname".**

На всех серверах системы желательно указать одинаковое имя пользователя (любое, кроме **ssw**). Лицензия ECSS-10 привязывается к ключу eToken/ruToken и к имени компьютера (**hostname**). Системный пользователь **ssw** создается при инсталляции пакета **ecss-user**.

✓ При установке системы в кластере, **рекомендуемое** значение hostname для первого сервера — **ecss1**, для второго — **ecss2**.

10.1.4 Проверка установки ОС

Проверка в основном сводится к правильности создания разделов диска и наличия доступа по ssh.

Для вывода информации о состоянии дискового пространства введите команду **df -h**. Она покажет общее и занятое место в разделах. Размеры разделов должны соответствовать проекту и введенным значениям при установке.

Для проверки доступа по ssh с машины, находящейся в одной подсети со вновь установленным сервером, нужно выполнить команду:

```
ssh <user>@<IP_ecss>
```

где:

- <user> — имя пользователя, заданного при установке;
- <IP_ecss> — IP-адрес хоста, заданного при установке.

10.1.5 Настройка /etc/hosts

Доменному имени хоста ecss1 должен соответствовать адрес 127.0.1.1. Также нужно прописать адрес хоста ecss2. Для этого в файле /etc/hosts необходимо прописать IP-адреса хостов ecss.

Например, для кластера: ecss1 имеет адрес 10.0.10.41, ecss2 — 10.0.10.42. Данные адреса нужно прописать в /etc/hosts:
посмотреть можно командой ***cat /etc/hosts***
изменить/добавить командой ***sudo nano /etc/hosts***
после модификации должен проходить пинг с сервера ecss1 ***ping ecss2*** и наоборот с сервера ecss2 ***ping ecss1***

на обоих хостах

```
sudo nano /etc/hosts
```

на ecss1

```
127.0.0.1 localhost
10.0.10.41 ecss1
10.0.10.42 ecss2
```

на ecss2

```
127.0.0.1 localhost
10.0.10.42 ecss2
10.0.10.41 ecss1
```

10.2 Конфигурация репозиториев

После установки для обновления будет использоваться только образ DVD диска, загрузка с репозитория отключена. Давайте разрешим обновление пакетов с репозитория Astra Linux.

на обоих хостах

```
sudo nano /etc/apt/sources.list
```

привести информацию в файле к следующему виду:

на обоих хостах

```
deb https://download.astralinux.ru/astra/frozen/1.8_x86-64/1.8.3/uu/1/extended-
repository/ 1.8_x86-64 main non-free non-free-firmware contrib
deb https://download.astralinux.ru/astra/frozen/1.8_x86-64/1.8.3/uu/1/main-repository/
1.8_x86-64 main contrib non-free non-free-firmware
#deb cdrom:[OS Astra Linux 1.8.1.12  DVD]/ 1.8_x86-64 contrib main non-free non-free-
firmware
```

Добавить репозиторий Eltex:

на обоих хостах

```
sudo sh -c "echo 'deb [arch=amd64] http://archive.eltex.org/ssw/1.8_x86-64/3.18 stable main
extras external' >> /etc/apt/sources.list.d/eltex-ecss10-stable.list"
```

Далее необходимо выполнить импорт ключа командой:

на обоих хостах

```
sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 33CB2B750F8BB6A5
```

10.2.1 Изменение приоритета репозиториев

Для установки Erlang, Nodejs и Elixir из репозитория Eltex, необходимо изменить приоритет репозиториев для данных пакетов.

Добавить в файл /etc/apt/preferences.d/erlang строки:

на обоих хостах

```
sudo sh -c "echo -e 'Package: erlang*\nPin: version 2:24.3*\nPin-Priority: 1000' >> /etc/apt/
preferences.d/erlang"
```

Результат: cat /etc/apt/preferences.d/erlang

```
Package: erlang*
Pin: version 2:24.3*
Pin-Priority: 1000
```

Добавить в файл /etc/apt/preferences.d/nodejs строки:

на обоих хостах

```
sudo sh -c "echo -e 'Package: nodejs*\nPin: version 14.20.0-1nodesource1*\nPin-Priority: 1000'
>> /etc/apt/preferences.d/nodejs"
```

Результат: cat /etc/apt/preferences.d/nodejs

```
Package: nodejs*
Pin: version 14.20.0-1nodesource1*
Pin-Priority: 1000
```

Добавить в файл /etc/apt/preferences.d/elixir строки:

на обоих хостах

```
sudo sh -c "echo -e 'Package: elixir*\nPin: version 1.14*\nPin-Priority: 1000' >> /etc/apt/preferences.d/elixir"
```

Результат: cat /etc/apt/preferences.d/elixir

```
Package: elixir*
Pin: version 1.14*
Pin-Priority: 1000
```

Выполнить обновление:

на обоих хостах

```
sudo apt update
```

Установить пакет ntp следующей командой:

на обоих хостах

```
sudo apt install ntp
```

10.2.2 "Оптимизация" Операционной Системы

Выставить параметры ОС в режим производительности

Используем утилиту cpufrequtils.

на обоих хостах

```
sudo apt install cpufrequtils
```

по умолчанию после инсталляции ОС использует режим "ondemand" - "по запросу" (производительность CPU по запросу приложений, экономит электроэнергию , но ниже производительность):

на обоих хостах

```
cat /etc/init.d/cpufrequtils | grep GOVERNOR=
```

в выходном сообщении системы , режим работы по умолчанию после инсталляции - "ondemand" :

```
GOVERNOR="ondemand"
```

Установить режим - результативность/производительность - в файле /etc/init.d/cpufrequtils значение "**ondemand**" заменить на "**performance**"

на обоих хостах

```
sudo sed -i 's/GOVERNOR="ondemand"/GOVERNOR="performance"/g' /etc/init.d/cpufrequtils
```

Перезапустить утилиту:

на обоих хостах

```
sudo /etc/init.d/cpufrequtils restart
```

Затем выполнить команду:

на обоих хостах

```
sudo systemctl daemon-reload
```

Отключить SWAP

Ubuntu сервер SSW работает **в реальном масштабе времени** ,
поэтому **все** необходимые **данные должны находится в оперативной памяти,** использование файла подкачки (swap-файл - /swap. img) может привести к увеличению времени обработки вызовов приложения SSW ECSS10 , **что недопустимо**. Swap - отключаем.
Выполнить последовательно три команды:

Отключение Swap:

на обоих хостах

```
sudo swapoff -a
```

Удалить файл swap.img.

на обоих хостах

```
sudo rm /swapfile
```

Удалить эту строчку (//swapfile swap swap defaults 0 0)

на обоих хостах

```
sudo nano /etc/fstab
```

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda2 during curtin installation
UUID=4c88cb4a-42c9-4b52-98c2-78da7e23b621 /                       ext4    defaults
1 1
```

Для проверки выполните команду **free -h:**

на обоих хостах

```
free -h
```

Размер Swap равен 0 - то есть он выключен

```
free -h
               total        used        free      shared  buff/cache   available
Mem:           3,9G        110M        3,2G        820K        535M        3,5G
Swap:            0B          0B          0B
```

Установка часового пояса

При инсталляции astara linux **не предлагается** установить часовой пояс(по умолчанию устанавливается время UTC). Его нужно установить **вручную** (для корректной работы системы тарификации , работ по расписанию и т.д.), например:

на обоих хостах

```
sudo timedatectl set-timezone Asia/Novosibirsk
```

В случае выноса компонентов системы на разные серверы/ВМ (ecss-node/ecss-msr) необходимо указывать единый часовой пояс, для всех компонентов системы.

Улучшение работы высоконагруженных серверов

Улучшить работу высоконагруженных серверов можно увеличив лимит открытых файлов.

Для установки лимита открытых файлов необходимо:

Проверить текущий лимит командой:

на обоих хостах

```
ulimit -a
```

результат :

```
eltex@ecss1:~$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
scheduling priority             (-e) 0
file size               (blocks, -f) unlimited
pending signals                 (-i) 15515
max locked memory       (kbytes, -l) 65536
max memory size         (kbytes, -m) unlimited
open files                      (-n) 1024
pipe size            (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
real-time priority              (-r) 0
stack size              (kbytes, -s) 8192
cpu time               (seconds, -t) unlimited
max user processes              (-u) 15515
virtual memory          (kbytes, -v) unlimited
file locks                      (-x) unlimited
```

Данного лимита (open files **1024**) **недостаточно** для нормальной работы высоконагруженных серверов.

Установить лимит открытых файл для каждого пользователя:

на обоих хостах

```
sudo sed -i  '55i\*               soft    nproc           65536\n*               hard
nproc           131072\n*               soft    nofile          65536\n*               hard
nofile          131072\nroot            -       memlock         unlimited' /etc/security/
limits.conf
```

Результат:

```
# /etc/security/limits.conf
#
#Each line describes a limit for a user in the form:
#
#<domain>        <type>  <item>  <value>
#
#Where:
```

```
#<domain> can be:
#        - a user name
#        - a group name, with @group syntax
#        - the wildcard *, for default entry
#        - the wildcard %, can be also used with %group syntax,
#                 for maxlogin limit
#        - NOTE: group and wildcard limits are not applied to root.
#          To apply a limit to the root user, <domain> must be
#          the literal username root.
#
#<type> can have the two values:
#        - "soft" for enforcing the soft limits
#        - "hard" for enforcing hard limits
#
#<item> can be one of the following:
#        - core - limits the core file size (KB)
#        - data - max data size (KB)
#        - fsize - maximum filesize (KB)
#        - memlock - max locked-in-memory address space (KB)
#        - nofile - max number of open files
#        - rss - max resident set size (KB)
#        - stack - max stack size (KB)
#        - cpu - max CPU time (MIN)
#        - nproc - max number of processes
#        - as - address space limit (KB)
#        - maxlogins - max number of logins for this user
#        - maxsyslogins - max number of logins on the system
#        - priority - the priority to run user process with
#        - locks - max number of file locks the user can hold
#        - sigpending - max number of pending signals
#        - msgqueue - max memory used by POSIX message queues (bytes)
#        - nice - max nice priority allowed to raise to values: [-20, 19]
#        - rtprio - max realtime priority
#        - chroot - change root to directory (Debian-specific)
#
#<domain>      <type>  <item>         <value>
#
```

```
#*               soft    core            0
#root            hard    core            100000
#*               hard    rss             10000
#@student        hard    nproc           20
#@faculty        soft    nproc           20
#@faculty        hard    nproc           50
#ftp             hard    nproc           0
#ftp             -       chroot          /ftp
#@student        -       maxlogins       4
*                soft    nproc           65536
*                hard    nproc           131072
*                soft    nofile          65536
*                hard    nofile          131072
root             -       memlock         unlimited
# End of file
```

10.3 Настройка сетевых интерфейсов

⚠ Получение адресов на сетевых интерфейсах по DHCP **недопустимо** на серверах ECSS.

Сетевые настройки необходимо выполнять с помощью Netplan.

По умолчанию в Astra Linux пакет netplan не установлен.

Установить пакеты для обслуживание функции netplan:

на обоих хостах

```
sudo apt install -y dialog libssl1.0.0 netplan.io
```

Сейчас можно выполнить конфигурацию netplan:

на обоих хостах

```
sudo nano /etc/netplan/ecss.yaml
```

10.3.1 Пример netplan

Netplan для интерфейсов сервера ecss1 (/etc/netplan/ecss.yaml)

на ecss1

```
# Netplan для хоста ecss1 программного коммутатора
# Обратите внимание на обязательное наличие минимум двух пробелов в каждой строке и секции (кроме строки секции network)

network:
  version: 2 # Версия netplan
  renderer: networkd # Исполнитель конфигураций netplan
  ethernets: # Раздел описания интерфейсов ethernet
    enp0s3: # Название внутреннего интерфейса ВМ
      dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
      addresses: [192.168.56.41/24] # Интерфейс управления OAM
      gateway4: 192.168.56.1
    enp0s8: # Название внешнего интерфейса ВМ
      dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
  vlans:
    net.10: # Интерфейс управления/Postgres
        id: 10
        link: enp0s8
        addresses: [10.0.10.41/24]# Адрес для Postgres
    net.20: # Интерфейс для VoIP
      id: 20
      link: enp0s8
      addresses: [10.0.20.41/24]# Адрес для VoIP / SIP
      routes:
        - to: 10.0.20.0/24
          via: 10.0.20.1
```

Netplan для интерфейсов сервера ecss2 (/etc/netplan/ecss.yaml)

на ecss2

```
# Netplan для хоста ecss2 программного коммутатора
# Обратите внимание на обязательное наличие минимум двух пробелов в каждой строке и секции (кроме строки секции network)

network:
  version: 2 # Версия netplan
  renderer: networkd # Исполнитель конфигураций netplan
  ethernets: # Раздел описания интерфейсов ethernet
    enp0s3: # Название внутреннего интерфейса ВМ
      dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
      addresses: [192.168.56.42/24] # Интерфейс управления OAM
      gateway4: 192.168.56.1
    enp0s8: # Название внешнего интерфейса ВМ
      dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
  vlans:
    net.10: # Интерфейс управления/Postgres
        id: 10
        link: enp0s8
        addresses: [10.0.10.42/24]# Адрес для Postgres
    net.20: # Интерфейс для VoIP
      id: 20
      link: enp0s8
      addresses: [10.0.20.42/24]# Адрес для VoIP / SIP
      routes:
        - to: 10.0.20.0/24
          via: 10.0.20.1
```

Так же проверить отсутствие в каталоге **/etc/netplan/** других файлов, если другие файлы присутствуют , то их нужно переместить в другой каталог или удалить, в противном случае возможна некорректная настройка сетевых интерфейсов и некорректная работа SSW.

на обоих хостах

```
ls -la /etc/netplan/
```

Применить, установленные параметры командой:

на обоих хостах

```
sudo netplan apply
```

Просмотреть получившиеся настройки можно при помощи команд **ifconfig** или **ip a**:

Для применения новых сетевых настроек необходимо выполнить команду **netplan apply.** Перезапуск сети или системы не требуется. Подробнее про настройки netplan см. в Приложении E. Netplan.

10.4 Обновление ОС и инсталляция необходимого ПО

10.4.1 Обновление системы

Перед началом установки **ecss** необходимо обновить ОС:

на обоих хостах

```
sudo apt update
```

на обоих хостах

```
sudo apt upgrade --enable-upgrade
```

Доступ между серверами без использования пароля

Далее следует разрешить доступ между серверами с через ssh по rsa-ключам - **без использования пароля**.

На хосте *ecss1 (затем на ecss2)* сгенерировать rsa-ключ следующий командой (выполнить **без sudo**, чтобы ключ был создан для текущего пользователя), на вопросы, использовать значение **по умолчанию**, нажав **три раза Enter**:

на обоих хостах

```
ssh-keygen
```

Скопировать rsa-ключ на сервер партнера ecss2 командой - **ssh-copy-id** ecss2.

на ecss1

```
ssh-copy-id ecss2
```

Скопировать rsa-ключ на сервер партнера ecss1 командой - **ssh-copy-id** ecss1.

на ecss2

```
ssh-copy-id ecss1
```

Проверить подключение между серверами, со стороны сервера ecss1 - **ssh ecss2** соединение должно быть установлено **без запроса пароля** . По окончанию теста вернутся обратно на сервер ecss1 , командой **exit.**

на ecss1

```
ssh ecss2
```

Аналогичным образом со стороны сервера ecss2 - **ssh ecss1** соединение должно быть установлено **без запроса пароля .** По окончанию теста вернутся обратно на сервер ecss2 , командой **exit**.

на ecss2

```
ssh ecss1
```

10.4.2 Инсталляция необходимого ПО

Установка дополнительных пакетов программ

Для работы SSW, потребуются дополнительные пакеты ПО не установленные в Astra Linux по умолчанию, но доступные в репозитории.

выполнить сначала на ecss1, затем на ecss2

```
sudo apt install ntp iptables tcpdump vlan dnsmasq aptitude atop ethtool htop mc minicom mtr-tiny nmap pptpd  screen ssh tftpd-hpa sngrep tshark  gnuplot debconf-utils keepalived attr bridge-utils ncdu wget
```

Пакеты ПО которые рекомендуется установить для работы

Список **обязательного** сервисного программного обеспечения:

```
sudo apt install ntp tcpdump vlan dnsmasq
```

ntp	NTP-сервер
tcpdump	сниффер пакетов
vlan	управление VLAN
dnsmasq	легковесный DNS/DHCP-сервер

Список **рекомендуемого** диагностического и вспомогательного программного обеспечения:

```
sudo apt install aptitude atop ethtool htop mc screen ssh tftpd sngrep tshark gnuplot
debconf-utils iotop ncdu
```

aptitude	установка программ из репозиториев, рекомендуется использовать вместо программы apt/apt-get
atop	мониторинг загрузки хоста с функцией периодического сохранения информации в файлы
ethtool	просмотр статистики сетевых интерфейсов
htop	мониторинг процессов
mc	файловый менеджер
screen	мультиплексор терминалов
ssh	сервер и клиент SSH
tftpd	TFTP-сервер
sngrep	трассировка sip
tshark	консольный аналог wireshark
gnuplot	вывод графиков статистики
debconf-utils	набор утилит для работы с базой debconf
iotop	инструмент для мониторинга использования ввода-вывода (IO) на диске в реальном времени в Linux
ncdu	утилита для поиска больших директорий в системе Linux

⚠ Данное программное обеспечение не требуется для работы системы ECSS-10, однако может упростить сервисное обслуживание системы и её отдельных компонентов со стороны инженеров эксплуатации и техподдержки.

Список **обязательных** пакетов для схем **с резервированием**:

```
sudo apt install ifenslave-2.6 keepalived attr
```

ifenslave-2.6	управление BOND-интерфейсами
keepalived	сервис мониторинга серверов/служб в кластере
attr	сервис управления атрибутами файловых систем

Список **дополнительных** пакетов для схем **с резервированием**:

```
sudo apt install bridge-utils ethtool
```

bridge-utils	управление bridge-интерфейсами
ethtool	управление и мониторинг сетевых интерфейсов

Для **просмотра** установленных пакетов выполнить следующую команду (**данный пункт является необязательным**: его можно выполнить, если не уверены, что какие-то пакеты установлены):

на обоих хостах

```
sudo dpkg --get-selections
```

⚠ Перед началом установки пакетов ecss , нужно убедится в соответствии полосы пропускания СПД необходимым требованиям.
Для этого выполнить команду **sudo ethtool <имя интерфейса>** для всех физических интерфейсов.
Пример:

```
sudo ethtool net.20
Settings for net.20:
    Supported ports: [ TP ]
    Supported link modes:   10baseT/Half 10baseT/Full
                            100baseT/Half 100baseT/Full
                            1000baseT/Full
    Supported pause frame use: No
    Supports auto-negotiation: Yes
    Supported FEC modes: Not reported
    Advertised link modes:  10baseT/Half 10baseT/Full
                            100baseT/Half 100baseT/Full
                            1000baseT/Full
    Advertised pause frame use: No
    Advertised auto-negotiation: Yes
    Advertised FEC modes: Not reported
    Speed: 1000Mb/s
    Duplex: Full
    Auto-negotiation: on
    Port: Twisted Pair
    PHYAD: 0
    Transceiver: internal
    MDI-X: off (auto)
    Link detected: yes
```

Проверить значение следующих параметров:
Advertised auto-negotiation: **Yes**
Speed: **1000Mb/s** (не менее)
Duplex: **Full**

10.5 Инсталляция пакетов ECSS

10.5.1 Предварительные требования

- Установленная и обновленная операционная система Astra Linux;
- Отсутствие в системе пользователя с именем **ssw**;
- Разбиение дискового пространства в соответствии с рекомендациями;
- Настроенная сеть;
- Установленный набор необходимых пакетов;

- Доступ к репозиторию ELTEX.

установка пакетов deb

```
   ┌ версия ПО ECSS. Сейчас в заявке на сертификацию она заявлена как 3.

   │ Мы её менять не будем до следующей сертификации или
  │ выпуска принципиально новой версии системы.

   │   ┌ версия System Release. Общая версия для всех компонент входящих в конкретный релиз.

   │      │ Меняем её централизовано,
   │   │ принимаем решение о выпуске нового релиза. Релизы между собой обычно не совместимы.
┌──┘ ┌──┘
ECSS.SysRel.SubMaj.SubMin
              └────┘ └────┘
        │   └ версия Subsystem minor. Минорную версию подсистемы устанавливает разработчик
подсистемы.

                  │          Минорную версию
        │     меняем при добавлении патча. Как правило, минорные версии в рамках одной мажорной
версии
        │     совместимы между собой и отличаются в рамках конкретных патчей.

                  └ версия Subsystem major. Мажорная версия устанавливается разработчиком
подсистемы.

                     Мажорную версию подсистемы необходимо менять при внесении в подсистему
существенных изменений.
```

Утилита установки пакетов APT анализирует версию пакета слева направо, для примера, мы имеем пакет 14.14.7.7 ,
в репозитории находятся пакеты
14.14.7.8
14.14.7.9
14.14.8.1
14.14.20 -14.14.28
то при выполнении команды **sudo apt install имя пакета** → будет автоматически проверен и установлен пакет 14.14.28 , так как он самый последний, анализ будет выполнен по 3-й позиции (major), анализ по 4-й (minor) выполнен не будет (аналогичным образом будет выполнятся команда **sudo apt upgrade --enable-upgrade**).
В случае если для конкретной ситуации требуется перейти с версии 14.14.7.7 на версию 14.14.7.9 , стандартная команда → **sudo apt upgrade --enable-upgrade** , нам не поможет, так как будет выбран самый новый пакет, в данной ситуации нам **необходимо в явном виде** указать какую версию пакета мы хотим установить, в данном примере мы **должны** выбрать команду → **sudo apt install имя пакета=14.14.7.9** . Обычно это необходимо для тестирования определенного патча, для стандартных обновлений достаточно выбора привычной команды установки/обновления пакета

В ходе установки пакетов ECSS нужно будет ответить на ряд вопросов для формирования необходимой конфигурации. Для автоматической загрузки требуемых настроек можно воспользоваться командами из пакета debconf-utils.
Описание и примеры использования при работе с debconf приведены в Приложении B. Debconf.

Для инсталляции системы ECSS-10 необходимо устанавливать пакеты в порядке, в котором они описаны ниже в документации.

10.5.2 Инсталляция обязательных пакетов ECSS

Установка пакета ecss-dns-env

Выполнить следующую команду:

на обоих хостах

```
sudo apt install -y ecss-dns-env
```

Настройщик предложит выбрать разделы для настройки по вопросам ниже. **Нужно выбрать broker.** (для примера 10.0.10.51 и 10.0.10.52)

Вопрос ecss-dns-env	Ответы для ecss1	Ответы для ecss2	Пример
[Primary broker] Введите адрес ([Primary broker] Enter ip)	**10.0.10.41** (необходимо ввести)	**10.0.10.41** (необходимо ввести)	
[Secondary broker] Введите адрес ([Secondary broker] Enter ip)	**10.0.10.42** (необходимо ввести)	**10.0.10.42** (необходимо ввести)	

ВАЖНО

В случае последующей коррекции IP-адресов следует использовать команду:

```
sudo dpkg-reconfigure ecss-dns-env
```

Настройка преобразование DNS имени в IP адрес

Для корректной работы системы, после установки пакета ecss-node требуется скорректировать дефолтные установки . Для этого:

В файле /etc/resolv.conf изменить параметр nameserver установленный системой по умолчанию на 127.0.0.1 (если строк nameserver несколько - остальные удалить/закомментировать):

```
sudo nano /etc/resolv.conf
```

Пример:

```
domain eltex.loc
search eltex.loc
nameserver 127.0.0.1
```

В Astra Linux после перезагрузки файл /etc/resolv.conf приводится к исходному виду, поэтому его **нужно сделать неизменяемым**:

```
sudo chattr +i /etc/resolv.conf
```

Для доступа к репозиториям AstraLinux и Элтекс нужно указать локальный DNS или глобальный (для примера 8.8.8.8 или 77.88.8.8)

Добавить в файл /etc/dnsmasq.d/ecss строки server=<Ipadd DNS сервера> (в примере 77.88.8.8) :

```
sudo sh -c "echo 'server=77.88.8.8' >> /etc/dnsmasq.d/ecss"
```

Перезапустить сервис:

```
sudo systemctl restart dnsmasq.service
```

Установка Postgres

Примечание

Перед установкой `ecss-postgres-bdr-ssw` в системе, в `apt sources,` должен быть добавлен `docker` для этого выполнить следующие команды:

```
sudo install -m 0755 -d /etc/apt/keyrings
```

```
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/
docker.asc
```

```
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

```
sudo sh -c "echo 'deb [arch=amd64 signed-by=/etc/apt/keyrings/docker.asc] https://
download.docker.com/linux/ubuntu   jammy stable' >> /etc/apt/sources.list.d/docker.list"
```

Выполнить обновление:

```
sudo apt update && sudo apt upgrade --enable-upgrade
```

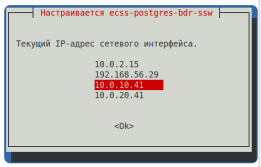
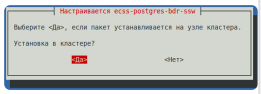
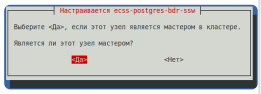
Установка пакета ecss-postgres-bdr-ssw

Выполняется последовательно, первым устанавливается на сервере который работает в режиме "Master"

ecss1

```
sudo apt install -y ecss-postgres-bdr-ssw
```

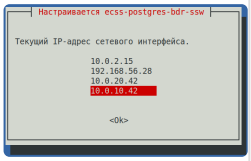
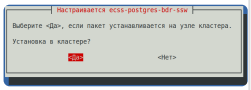
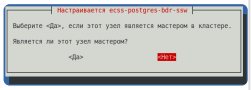
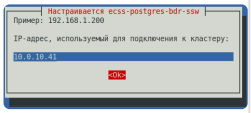
При конфигурации пакета **ecss-postgres-bdr-ssw** будут заданы вопросы:

Вопросы ecss-postgres-bdr-ssw	Ответы для ecss1 (Master)	Пример
Текущий IP-адрес сетевого интерфейса.	**Выбрать** из списка IP-адрес для работы с Postgres	
Установка в кластере?	Yes (значение по умолчанию)	
Является ли этот узел мастером?	**Yes** (необходимо выбрать)	

ecss2

```
sudo apt install -y ecss-postgres-bdr-ssw
```

При конфигурации пакета **ecss-postgres-bdr-ssw** будут заданы вопросы:

Вопросы ecss-postgres-bdr-ssw	Ответы для ecss2 (Slave)	Пример
Текущий IP-адрес сетевого интерфейса.	**Выбрать** из списка IP-адрес для работы с Postgres	
Установка в кластере?	Yes (значение по умолчанию)	
Является ли этот узел мастером?	No (значение по умолчанию)	
IP-адрес, используемый для подключения к кластеру:	**Указать** IP-адрес Master (ecss1)	

Проверить корректность настройки репликации

Во время установки в директорию `/srv/ecss/ecss-postgres-bdr-ssw` устанавливаются скрипт **postgresbdr_ssw.sh**

Для проверки репликации выполнить следующие команды:

Для проверки репликации выполнить команду на обоих хостах:

```
cd /srv/ecss/ecss-postgres-bdr-ssw
```

Для проверки репликации выполнить команду на обоих хостах:

```
sudo ./postgresbdr_ssw.sh check
```

Вывод на хостах **должен быль одинаковый:**

ecss1

```
sudo ./postgresbdr_ssw.sh check
Enter master ip please
Like 10.150.150.15
10.0.10.41
Enter slave ip please
10.0.10.42
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
таблица нод на слейве по ecss_storekeeper_db
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                              conn_dsn

---------------------
+-----------------------------------------------------------------------------
----------
 7577617967861174292 | host=10.0.10.41 port=5439 dbname=ecss_storekeeper_db
```

user=postgres password=postgres1
 7577618825220177940 | host=10.0.10.42 port=5439 dbname=ecss_storekeeper_db user=postgres password=postgres1
(2 строки)

таблица коннекций на слейве по ecss_storekeeper_db
     conn_sysid | conn_dsn

+---

 7577617967861174292 | host=10.0.10.41 port=5439 dbname=ecss_storekeeper_db user=postgres password=postgres1
 7577618825220177940 | host=10.0.10.42 port=5439 dbname=ecss_storekeeper_db user=postgres password=postgres1
(2 строки)

ecss2

sudo ./postgresbdr_ssw.sh check
Enter master ip please
Like 10.150.150.15
10.0.10.41
Enter slave ip please
10.0.10.42
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
таблица нод на слейве по ecss_storekeeper_db
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid | conn_dsn

+---

 7577617967861174292 | host=10.0.10.41 port=5439 dbname=ecss_storekeeper_db

```
user=postgres password=postgres1
 7577618825220177940 | host=10.0.10.42 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 строки)

таблица коннекций на слейве по ecss_storekeeper_db
     conn_sysid       |                                      conn_dsn

---------------------
+-----------------------------------------------------------------------------
----------
 7577617967861174292 | host=10.0.10.41 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7577618825220177940 | host=10.0.10.42 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 строки)
```

пример с ошибками

нет доступа до slave

```
sudo ./postgresbdr_ssw.sh check
[sudo] password for abf:
Sorry, try again.
[sudo] password for abf:
Enter master ip please
Like 10.150.150.15
10.0.10.51
Enter slave ip please
10.0.10.52
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
     node_sysid       |                                  node_local_dsn
                                  | node_init_from_dsn
---------------------
+---------------------------------------------------------------------------
-------------+--------------------
```

```
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1 |
(1 row)

таблица нод на слейве по ecss_storekeeper_db
psql: error: connection to server at "10.0.10.52", port 5439 failed:
Connection refused
    Is the server running on that host and accepting TCP/IP connections?
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                        conn_dsn
                                                                    |
---------------------
+-------------------------------------------------------------------------
-------------
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(1 row)

таблица коннекций на слейве по ecss_storekeeper_db
psql: error: connection to server at "10.0.10.52", port 5439 failed:
Connection refused
    Is the server running on that host and accepting TCP/IP connections?
```

ошибка несоответствия коннекций

```
sudo ./postgresbdr_ssw.sh check
Enter master ip please
Like 10.150.150.15
10.0.10.51
Enter slave ip please
10.0.10.52
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
     node_sysid      |                                        node_local_dsn
                                    | node_init_from_dsn
---------------------
+-------------------------------------------------------------------------
```

```
-------------+--------------------
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1 |
(1 row)
```

```
таблица нод на слейве по ecss_storekeeper_db
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+---------------------------------------------------------------------------
-------------
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(1 row)
```

```
таблица коннекций на слейве по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+---------------------------------------------------------------------------
-------------
 7569499877813944343 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.52 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(3 rows)
```

Установка ecss-node

Установка обязательного пакета **ecss-node** включает в себя установку и первоначальную настройку основных подсистем.

> ⚠ В системе уже должен быть установлен пакет **ecss-postgres-bdr-ssw**

Для установки пакета **ecss-node** выполните команду:

первоначально на ecss1

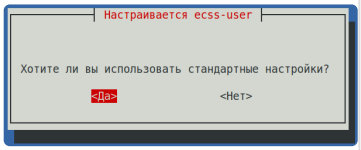
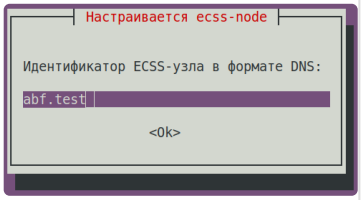
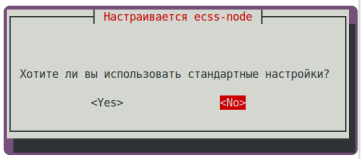
```
sudo apt install -y ecss-node
```

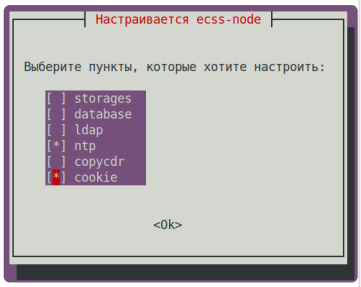
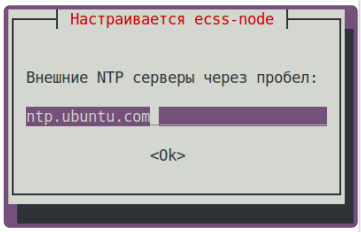
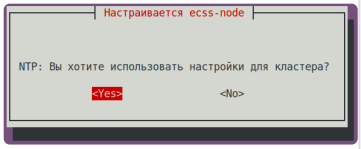
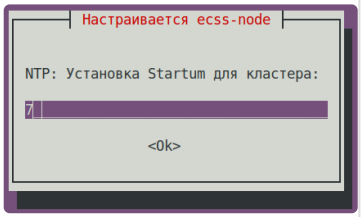
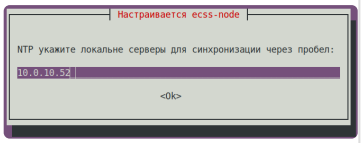
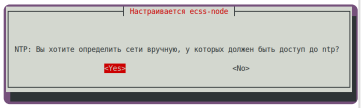
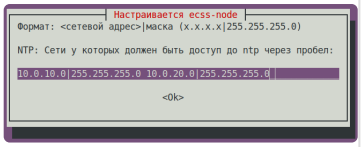
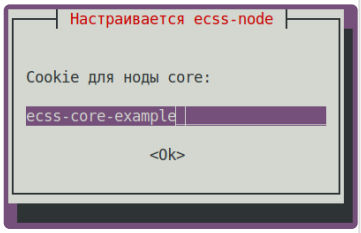
Во время установки пакета создается пользователь **ssw**, от имени которого запускаются все сервисы ecss*. Создаются необходимые каталоги, выполняется настройка DNS, идет настройка SSL-сертификатов.

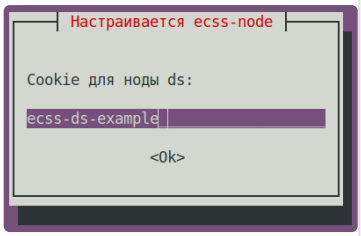
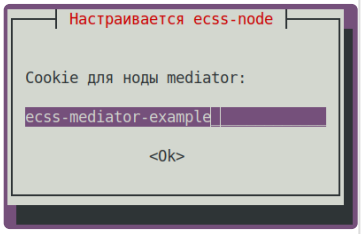
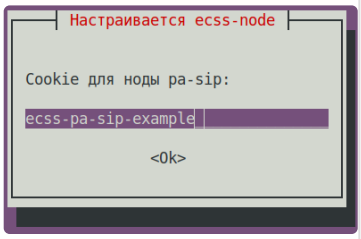
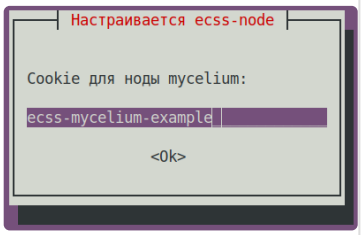
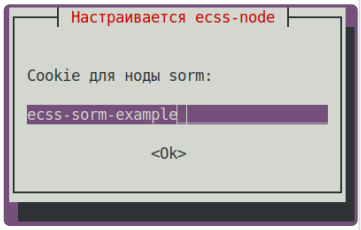
При установки так же будет установлен пакет **ecss-user**

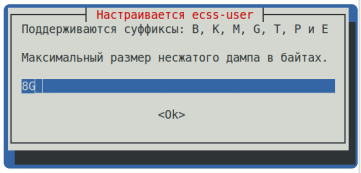
Установка на сервере **ecss1**

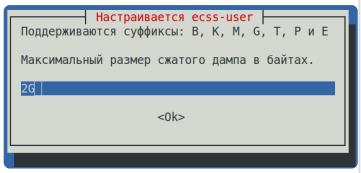
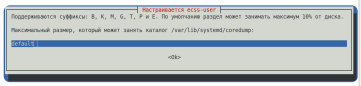
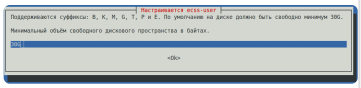
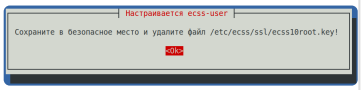
Во время установки будет предложено настроить параметры, необходимые для формирования конфигурационных файлов, примеры ответов следует ниже.

Вопросы ecss-user	**Ответы для ecss1**	**Пример**
Хотите ли вы использовать стандартные настройки?	Yes (значение по умолчанию)	
Вопросы ecss-node	**Ответы для ecss1**	**Пример**
Идентификатор ECSS-узла в формате DNS (Настройка имени кластера): (любое уникальное имя, в примере abf.test)	**abf.test** (необходимо ввести) Формат **ECSS_ID** Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире. `[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]`	
Хотите ли вы использовать стандартные настройки?	**No** (необходимо выбрать)	

Выберите пункты, которые хотите настроить:	**ntp + cookie** (необходимо выбрать)	
Внешние NTP серверы через пробел:	ntp.ubuntu.com (значение по умолчанию)	
NTP: Вы хотите использовать настройки для кластера?	Yes (значение по умолчанию)	
NTP: Установка Startum для кластера:	7 (значение по умолчанию)	
NTP укажите локальне серверы для синхронизации через пробел: (в примере ecss2 - 10.0.10.42)	**10.0.10.42** (необходимо ввести)	
NTP: Вы хотите определить сети вручную, у которых должен быть доступ до ntp?	Yes (значение по умолчанию)	
NTP: Сети у которых должен быть доступ до ntp через пробел: (в примере 10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0)	**10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0** (необходимо ввести)	
Cookie для ноды core: (указать уникальный cookie для core , в примере ecss-core-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-core-example** (необходимо ввести)	

Cookie для ноды ds: (указать уникальный cookie для ds , в примере ecss-ds-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-ds-example** (необходимо ввести)	
Cookie для ноды mediator: (указать уникальный cookie для mediator , в примере ecss-mediator-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mediator-example** (необходимо ввести)	
Cookie для ноды pa-sip: (указать уникальный cookie для pa-sip , в примере ecss-pa-sip-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-pa-sip-example** (необходимо ввести)	
Cookie для ноды mycelium: (указать уникальный cookie для mycelium , в примере ecss-mycelium-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mycelium-example** (необходимо ввести)	
Cookie для ноды sorm: (указать уникальный cookie для sorm , в примере ecss-sorm-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-sorm-example** (необходимо ввести)	

Вопросы ecss-user	**Ответы для ecss1**	**Пример**
Максимальный размер несжатого дампа в байтах.	8G (значение по умолчанию)	

Максимальный размер сжатого дампа в байтах.	2G (значение по умолчанию)	
Максимальный размер, который может занять каталог /var/lib/systemd/coredump:	default (значение по умолчанию)	
Минимальный объем свободного дискового пространства в байтах.	30G (значение по умолчанию)	
Сохраните в безопасное место и удалите файл /etc/ecss/ssl/ecss10root.key!	Ok (значение по умолчанию)	

После установки пакета ecss-node на хосте ecss1, выполните команду **"sudo usermod -a -G ssw <Имя_пользователя>" на хосте ecss1**.

Пример:

```
sudo usermod -a -G ssw eltex
```

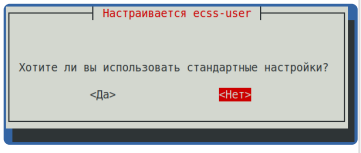
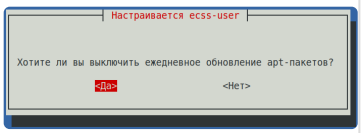
Затем **выполите установку на хосте ecss2.**
Используйте режим "Скопировать существующие сертификаты (copy) по ssh" с сервера ecss1

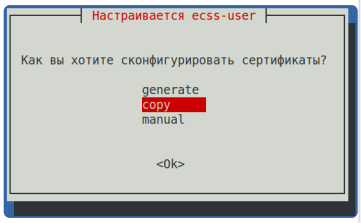
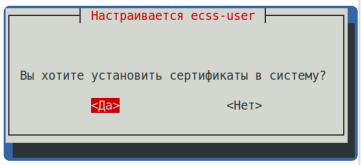
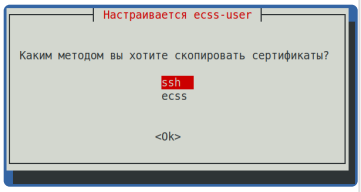
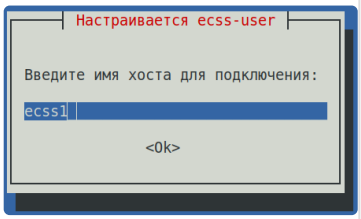
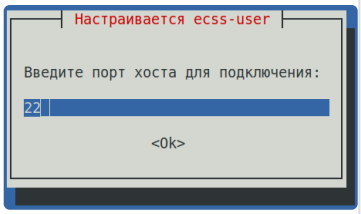
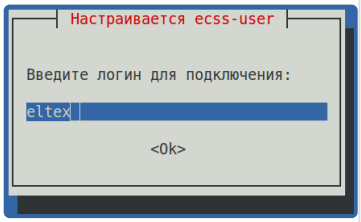
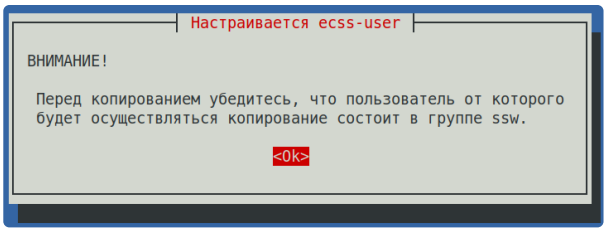
Установка на сервере **ecss2**

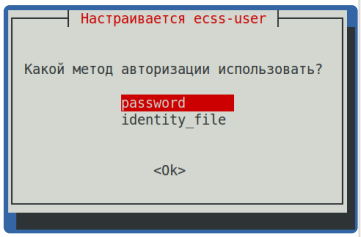
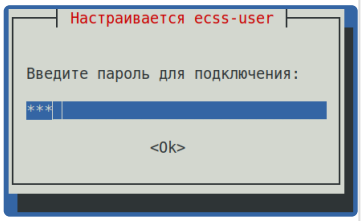
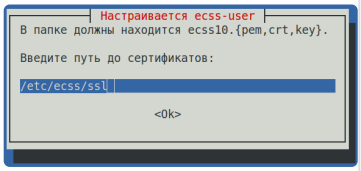
на ecss2

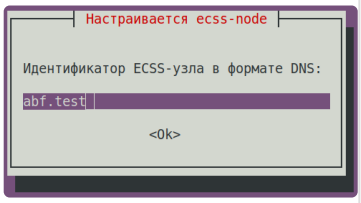
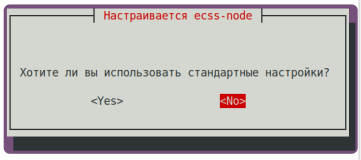
```
sudo apt install -y ecss-node
```

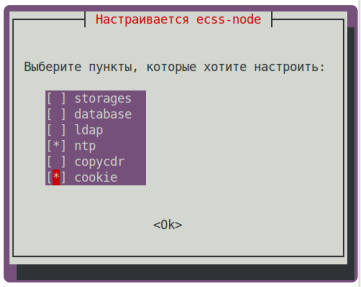
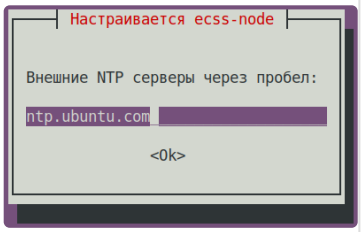
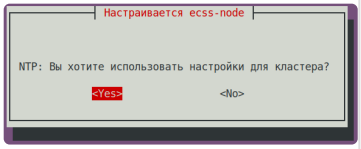
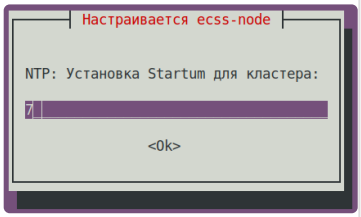
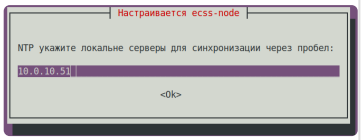
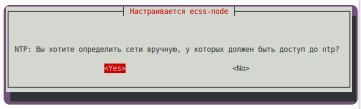
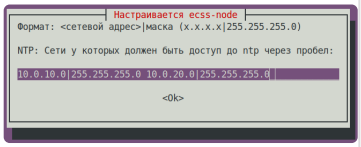
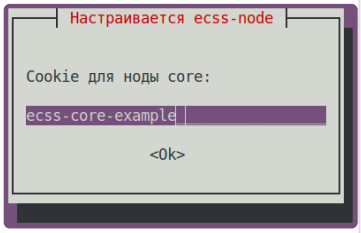
Во время установки будет предложено настроить параметры, необходимые для формирования конфигурационных файлов, примеры ответов следует ниже.

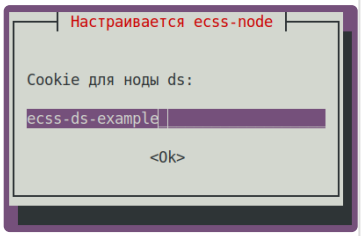
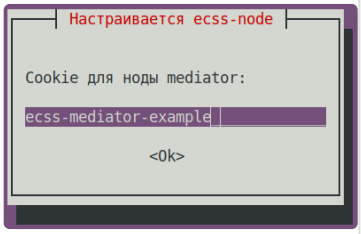
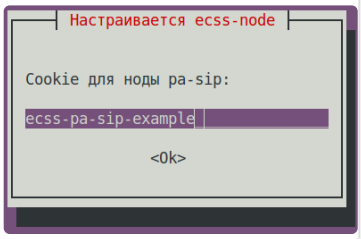
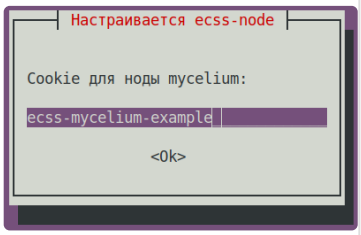
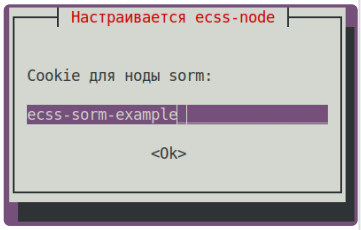
Вопросы ecss-user	Ответы для ecss2	Пример
Хотите ли вы использовать стандартные настройки?	No (значение по умолчанию)	
Хотите ли вы выключить ежедневное обновление apt-пакетов?	Yes (значение по умолчанию)	

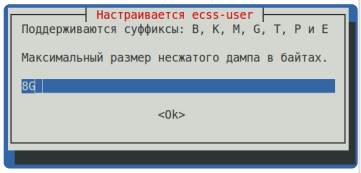
Как вы хотите сконфигурировать сертификаты?	**copy** (необходимо выбрать)	
Вы хотите установить сертификаты в систему?	Yes (значение по умолчанию)	
Каким методом вы хотите скопировать сертификаты?	**ssh** (необходимо выбрать)	
Введите имя хоста для подключения:	**ecss1** (необходимо ввести)	
Введите порт хоста для подключения:	22 (значение по умолчанию)	
Введите логин для подключения: (для примера eltex)	**eltex** (необходимо ввести)	
		

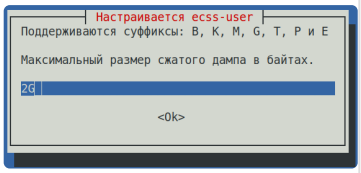
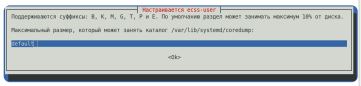
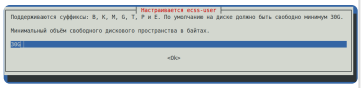
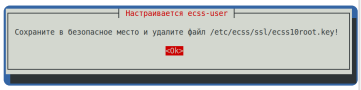
Какой метод авторизации использовать?	**password** (необходимо выбрать)	
Введите пароль для подключения:		
Введите путь до сертификатов:	/etc/ecss/ssl (значение по умолчанию)	

Вопросы ecss-node	**Ответы для ecss2**	**Пример**
Идентификатор ECSS-узла в формате DNS (Настройка имени кластера): (любое уникальное имя, в примере abf.test)	**abf.test** (необходимо ввести) Формат **ECSS_ID** Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире. `[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]`	
Хотите ли вы использовать стандартные настройки?	**No** (необходимо выбрать)	

Выберите пункты, которые хотите настроить:	**ntp + cookie** (необходимо выбрать)	
Внешние NTP серверы через пробел:	ntp.ubuntu.com (значение по умолчанию)	
NTP: Вы хотите использовать настройки для кластера?	Yes (значение по умолчанию)	
NTP: Установка Startum для кластера:	7 (значение по умолчанию)	
NTP укажите локальне серверы для синхронизации через пробел: (в примере ecss2 - 10.0.10.41)	**10.0.10.41** (необходимо ввести)	
NTP: Вы хотите определить сети вручную, у которых должен быть доступ до ntp?	Yes (значение по умолчанию)	
NTP: Сети у которых должен быть доступ до ntp через пробел: (в примере 10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0)	**10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0** (необходимо ввести)	
Cookie для ноды core: (указать уникальный cookie для core , в примере ecss-core-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-core-example** (необходимо ввести)	

Cookie для ноды ds: (указать уникальный cookie для ds , в примере ecss-ds-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-ds-example** (необходимо ввести)	
Cookie для ноды mediator: (указать уникальный cookie для mediator , в примере ecss-mediator-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mediator-example** (необходимо ввести)	
Cookie для ноды pa-sip: (указать уникальный cookie для pa-sip , в примере ecss-pa-sip-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-pa-sip-example** (необходимо ввести)	
Cookie для ноды mycelium: (указать уникальный cookie для mycelium , в примере ecss-mycelium-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mycelium-example** (необходимо ввести)	
Cookie для ноды sorm: (указать уникальный cookie для sorm , в примере ecss-sorm-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-sorm-example** (необходимо ввести)	

Вопросы ecss-user	**Ответы для ecss2**	**Пример**
Максимальный размер несжатого дампа в байтах.	8G (значение по умолчанию)	

Максимальный размер сжатого дампа в байтах.	2G (значение по умолчанию)	
Максимальный размер, который может занять каталог /var/lib/systemd/coredump:	default (значение по умолчанию)	
Минимальный объем свободного дискового пространства в байтах.	30G (значение по умолчанию)	
Сохраните в безопасное место и удалите файл /etc/ecss/ssl/ecss10root.key!	Ok (значение по умолчанию)	

Проверить статус сервисов следующей командой:

на обоих хостах

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
```

В случае статуса "active" продолжаем дальше, в случае статуса "inactive" выполнить команду "sudo systemctl restart ecss-<имя пакета>"

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
active
active
active
active
active
```

Установка лицензии

Система ECSS-10 использует ELM-лицензирование – сервис распределения лицензий ECSS-экосистемы, состоит из **ecss-license-agent** и **ecss-license-master** .

- **ECSS License Agent** - Адаптер для работы с EcssLicenseMaster (LM). Получает список хостов EcssLicenseMaster, реализует логику подключения и переподключения к LM.
  - Отвечает за получение, валидацию и дальнейшую отправку лицензий клиенту.
  - Отправляет статусы подключения и полученные лицензии клиенту.
  - Предоставляет API для отправки HTTP запросов и WS событий на LM.
- **Ecss License Master** - Сервис оркестрации параметров лицензии для ECSS-узлов.
  - Отвечает за:
    - Получение лицензии для конкретного ECSS-узла (SSW / AuP) либо устройства (SMG)
    - Распределение параметров лицензии между ECSS-узлами и ECSS-сервисами
    - Контроль активности ECSS-узлов и их валидация во избежание тиражирования лицензии сверх указанных лимитов

- Контроль изменения указанных лимитов на ECSS-узлах, и их динамическое распределение в соответствии с текущей нагрузкой.

ecss-license-agent является составной частью SSW, входит в пакет ecss-node. Отдельной конфигурации не требует.

В свою очередь **Ecss License Master** может быть установлен как на SSW так и на отдельно выделенный сервер. В случае если Ecss License Master уже установлен на вашем проекте, то потребуется только дополнить запись о новом компоненте в конфигурационном файле LM, и следующий пункт – установка "Установка LM" не потребуется.

Установка LM на SSW host

Для установки пакета **ecss-license-master** необходимо добавить репозиторий следующей командой:

если кластер, то на обоих хостах

```
sudo sh -c "echo deb [arch=amd64] https://archive.eltex.org/ssw/1.8_x86-64/aup/2.0 stable main extras external >> /etc/apt/sources.list.d/lm-stable.list"
```

Выполнить обновление из добавленного репозитория командой:

если кластер, то на обоих хостах

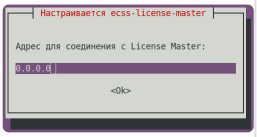
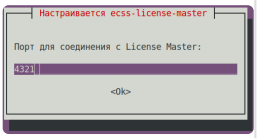
```
sudo apt update
```

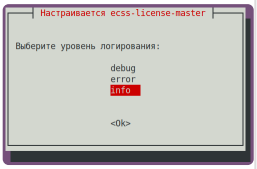
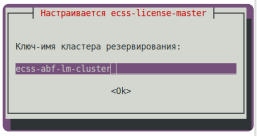
Установить пакет LM (ecss-license-master)

если кластер, то на обоих хостах

```
sudo apt install ecss-license-master
```

При установке пакета ecss-license-master будут заданы вопросы:

Вопросы	Ответы	Примеры
Адрес для соединения с License Master: License Master listen ip:	0.0.0.0 (значение по умолчанию)	
Порт для соединения с License Master: License Master listen port:	4321 (значение по умолчанию)	

Вопросы	Ответы	Примеры
Выберите уровень логирования: Log level:	info (значение по умолчанию), возможные варианты: debug, error, info	
Ключ-имя кластера резервирования: License Master cluster key:	указать уникальное имя ecss-lm-cluster (значение по умолчанию) Менеджеры с одинаковыми именами попытаются собраться в кластер и резервировать друг друга.	

Данные сохраняются в файле - **/etc/ecss/ecss-license-master/config.env** :

```
ECSS_LM_HOST=0.0.0.0
ECSS_LM_PORT=4321
ECSS_LM_LOG_LEVEL=info
RELEASE_COOKIE=ecss-abf-lm-cluster
```

Конфигурация LM

Сервис ecss-license-master использует два конфигурационных файла: **/etc/ecss/ecss-license-master/config.env** и **/etc/ecss/ecss-license-master/config.yaml**.
В файл **config.env** информация сохраняется в момент установки пакета **ecss-license-master**.
Config.yaml файл содержит основные настройки для подключения к ELM серверу, откуда брать лицензии, какие лицензии, и куда их потом передавать, **все это нужно сконфигурировать вручную**.

При установке LM на SSW

Выполнить команду:

если кластер, то на обоих хостах

```
sudo nano /etc/ecss/ecss-license-master/config.yaml
```

информация в файле после установки

```
elm_addresses: []
licenses: []
ecss_nodes: []
```

В поле **elm_adresses** нужно прописать список серверов ELM к которым мы хотим подключаться для получения лицензий. Первый адрес в списке используется как основной. Дополнительные строки с адресами работают как резерв.

Данные ELM сервера(ов) получить у менеджера проекта.

пример:
- "elm-1.eltex.loc:8099"
- "192.168.111.22:8099"
- "elm-3.eltex.loc:8099"

Дальше задаются сами лицензии в поле **licenses**.

- **id** – может быть любой, главное чтоб был уникальный у каждой лицензии. Влияет только на адрес лицензии на самом LM (например 0,1,2,3).
- **kind** – для SSW поддерживается только ***ssw.***
- **type** – выбирается как ***elm*** если запрашивать данную лицензию нужно с ELM сервера. Если нужно использовать файловую лицензию SSW, то тип выбирается как ***ecss_license***.
  - при ***elm*** типе, необходимо задать поля **license_key** и **product_id** для идентификации лицензии.
  - при ***ecss_license*** типе, задается поле **license** содержащее ключ статичной лицензии

Параметры **license_key** и **product_id** должны быть согласованы с данными загруженными на ELM сервере .

Значение параметров **license_key**, **product_id** или **license**, **passport** нужно получить у менеджера проекта.

Блок **ecss_nodes**.
В нем нужно задать данные сервисов которые будут получать лицензию. В нашем случае SSW.
Сервис задается тремя параметрами:

- **id –** идентификатор сервиса, должен совпадать с ECSS_ID лицензируемого SSW.
- **kind –** тип лицензируемого сервиса. по аналогии с блоком лицензий, для SSW поддерживается только ***ssw.***
- **license_id –** идентификатор лицензии которая будет отправляться на данный сервис. Соответствует идентификаторам лицензии из блока **licenses**

Два SSW с **разными ECSS_ID** не могут получить одну и ту же лицензию, конфигурация с таким распределением будет помечена некорректной.
Кроме того, два SSW с **одинаковыми ECSS_ID** не смогут одновременно получать одну и ту же лицензию с одного LM,
поэтому крайне важно корректно устанавливать свой ECSS_ID и делать его уникальным как минимум в зоне видимости используемого LM.

Формат **ECSS_ID**
Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире.

```
[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]
```

⚠ Для SSW работающего в кластере указываем только одну запись в блок **ecss_nodes** на кластер, индивидуально для каждого хоста указывать нельзя. Иначе будут генерироваться два запроса с одинаковым ECSS_ID что недопустимо.

Пример: файла config.yaml

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
   kind: SSW
   type: elm
   license_key: "ssw1test"
   product_id: "ECSS1000001"
 - id: 1
   kind: ssw
   type: ecss_license
   license: "af615ebb92d381125ff"
ecss_nodes:
 - id: abf.test
   kind: ssw
   license_id: 0
```

Выполнить перезагрузку сервиса **ecss-license-master.service** для обновления конфигурации из файла **/etc/ecss/ecss-license-master/config.yaml** следующей командой:

если кластер, то на обоих хостах

```
sudo systemctl restart ecss-license-master.service
```

При подключении SSW к ранее установленному LM

В случае если LM уже установлен и используется в локальной сети требуется добавить параметры нового SSW в конфигурационный файл **/etc/ecss/ecss-license-master/config.yaml** в блоках licenses и ecss_nodes.

Выполнить команду:

если кластер, то на обоих хостах

```
sudo nano /etc/ecss/ecss-license-master/config.yaml
```

для примера:

Было:

```
elm_addresses:
```

```
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
   kind: SSW
   type: elm
   license_key: "ssw1test"
   product_id: "ECSS1000001"
 - id: 1
   kind: ssw
   type: ecss_license
   license: "af615ebb92d381125ff"
 - id: 2
   kind: SSW
   type: elm
   license_key: "ssw2test"
   product_id: "ECSS1000002"
ecss_nodes:
 - id: 1.test
   kind: ssw
   license_id: 0
 - id: 2.test
   kind: ssw
   license_id: 1
 - id: 3.test
   kind: ssw
   license_id: 2
```

Стало:

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
   kind: SSW
   type: elm
   license_key: "ssw1test"
   product_id: "ECSS1000001"
 - id: 1
   kind: ssw
   type: ecss_license
   license: "af615ebb92d381125ff"
 - id: 2
   kind: SSW
   type: elm
   license_key: "ssw2test"
   product_id: "ECSS1000002"
 - id: 3
   kind: SSW
   type: elm
   license_key: "ssw3test"
   product_id: "ECSS1000003"
ecss_nodes:
 - id: 1.test
```

```
    kind: ssw
    license_id: 0
  - id: 2.test
    kind: ssw
    license_id: 1
  - id: 3.test
    kind: ssw
    license_id: 2
  - id: abf.test
    kind: ssw
    license_id: 3
```

Выполнить перезагрузку сервиса **ecss-license-master.service** для обновления конфигурации из файла **/etc/ecss/ecss-license-master/config.yaml** следующей командой:

если кластер, то на обоих хостах

```
sudo systemctl restart ecss-license-master.service
```

Подключение SSW к Менеджера Лицензий

Выполнить конфигурацию подключения SSW к LM следующей командой в CoCon, (в примере LM установлен в кластере, IPadd хостов 10.0.10.41 и 10.0.10.42. значение порта из файла /etc/ecss/ecss-license-master/config.env по умолчанию 4321):

выполняется в CoCon

```
/system/licence/manager/set --hosts [https://10.0.10.41:4321, https://10.0.10.42:4321]
```

В случае установки LM на хостах SSW указывать IP адреса OAM

После выполнения проверить статус подключения командой:

выполняется в CoCon

```
/system/licence/manager/show-status
┌──────────────────────────┬───────┬─────┐
│          Host            │Current│Alive│
├──────────────────────────┼───────┼─────┤
│https://10.0.10.41:4321   │*      │true │
│https://10.0.10.42:4321   │       │true │
└──────────────────────────┴───────┴─────┘
```

Все хосты должны иметь статус alive=true. Один должен быть "current".

- Current - отображает к какому хосту подключён SSW. Если есть подключение, указывается *, если нет - ничего не указано.
- Alive - отображает статус доступности хоста (healthcheck).

Подключение к License Manager

Отсутствие подключения к License Manager \ ELM будет равноценно отсутствию доступа к токену, что приведёт к критическим авариям, переходу в аварийный режим по истечению которого, если связь не будет восстановлена, система перейдёт на дефолтную лицензию.

данное поведение применимо только при работе в **elm** режиме

Если статус к одному из хостов LM показан current, можно посылать запрос на загрузку лицензии на SSW командой:

выполняется в CoCon

```
/cluster/storage/ds1/licence/request
```

пример выполнения команды /cluster/storage/ds1/licence/request

type=elm

```
/cluster/storage/ds1/licence/request
Licence received
[*******                                                    ] 6s 2ms
Success: Licence parameters applied
```

В режиме "type=ecss_license" команда **/cluster/storage/ds1/licence/request** выполнится только в случае если был предварительно загружен паспорт.

type=ecss_license

```
/cluster/storage/ds1/licence/request
Waiting for licence...
[*******                                                    ] 6s
┌─┬──────────────────────────────────────────────────────┬─────────┬──────────────────────────────────────────────┐
│A│                     Description                      │Old Value│                 New Value                    │
├─┼──────────────────────────────────────────────────────┼─────────┼──────────────────────────────────────────────┤
│^│Support for reservation of call-processes             │false    │true                                          │
│^│Maximum call duration (in seconds)                    │60       │2678400                                       │
│^│Maximum number of simultaneous calls                  │5        │10000                                         │
│^│Elph                                                  │         │                                              │
│^│ total count of members of the one Elph group         │10       │100                                           │
│^│Subscribers limit                                     │10       │500000                                        │
│^│Virtual subscribers limit                             │infinity │1000                                          │
```

```
|^|Add-on conferences                                  |         |
                |
|^| total count                                        |30       |1000
                |
|^|Add-on conferences                                  |         |
                |
|^| members of the one add-on conference               |16       |300
                |
|^|Chat rooms                                          |         |
                |
|^| total count                                        |30       |100
                |
|^|Chat rooms                                          |         |
                |
|^| members of the one chatroom                        |16       |300
                |
|^|Meet Me                                             |         |
                |
|^| total count                                        |0        |100
                |
|^|Meet Me                                             |         |
                |
|^| members of the one Meet Me conference              |0        |300
                |
|^|SORM                                                |         |
                |
|^| enabled                                            |false    |true
                |
|+| channels on SORM mediator                          |         |64
                |
|^|Sorm extractor                                      |none     |mfi
                |
|^|Support antifraud system                            |none     |custom
                |
|+|Call center                                         |         |
                |
|+| operator's capability for look at calls in queue   |         |true
                |
|+| operator's capability for use Intervension SS      |         |true
                |
|^|Call center                                         |         |
                |
|^| active agents                                      |0        |1000
                |
|^|Call center                                         |         |
                |
|^| active supervisors                                 |0        |100
                |
|^|Teleconference                                      |         |
                |
|^| channels                                           |3        |2000
                |
|^|Teleconference                                      |         |
                |
|^| max members of one teleconference                  |0        |200
                |
|^|Teleconference                                      |         |
                |
```

```
|^| active count                                          |0        |32
              |
|^|TSMN system                                           |         |
              |
|^| concurrent calls (active)                            |0        |50
              |
|^|TSMN system                                           |         |
              |
|^| concurrent calls (backup)                            |0        |50
              |
|^|TSMN system                                           |         |
              |
|^| backup mode                                          |none     |"backup"
              |
|^|Total count of simultaneous records voice calls       |0        |200
            |
|+|IVR                                                   |         |
              |
|+| customization enabled                                |         |true
              |
|+| Automatic Speech Recognition (ASR) subsystem enabled|         |true
              |
|^|Channels on dialer outgoing calls                     |2        |5
              |
|+|Supplementary Services licence package(s)            |         |
              |
|+| name                                                 |         |'ECSS-ADV'
              |
|+| limit                                                |         |10000
            |
|+| description                                          |         |"Additional services of 4
and 5 levels" |
|+| SS list                                              |         |[1,2,3,4,5]
              |
|+| name                                                 |         |'ECSS-BAS'
              |
|+| limit                                                |         |10000
            |
|+| description                                          |         |"Basic services of 1 and
2 levels"      |
|+| SS list                                              |         |[1,2]
              |
|+| name                                                 |         |'ECSS-BAS+'
              |
|+| limit                                                |         |10000
            |
|+| description                                          |         |"Expansion of basic
services of 3 level"|
|+| SS list                                              |         |[1,2,3]
              |
|^|GEO backup                                           |         |
              |
|^| enabled                                              |false    |true
              |
|^|SIGTRAN                                               |         |
              |
|^| SEP                                                  |false    |true
              |
```

```
|^| STP                                                    |false    |true
                |
└─┴──────────────────────────────────────────────────────────┴─────────┴──────────────────────────
─────────────────┘
Legend:
    '^' - Changed;
    '-' - Deleted;
    '+' - New.


[request] You are trying to add a license that is different from the current one by the pa
          rameters in the table (other values will remain unchanged).
Licence received      o ?> yes
[*******                                                              ] 1m 15s
Success: Licence parameters applied
```

В зависимости от выбранного типа лицензии для SSW : "type=**elm**" или "type=**ecss_license**" на команду просмотра лицензий получим разный вывод:

type=ecss_license

```
/cluster/storage/ds1/licence/list-licence
┌──┬──────────────────────┬──────────┬──────┬──────────────────┬──────────────────────┬──────────
──────────────────┬──────────┐
|Id|Creation date(UTC) |   SSW ID   |Active|  Description   |Comm. Exp.
Date(UTC)|Expiration date(UTC)|Time left|
├──┼──────────────────────┼──────────┼──────┼──────────────────┼──────────────────────┼──────────
──────────────────┼──────────┤
|1 |12.03.2026 18:40:00|ECSS 010070 |*     |ECSS TPM License|13.03.2027 06:59:59
|14.03.2026 10:44:05 |1d 6h 59m|
|0 |01.01.1990 00:00:00|ECSS DEFAULT|      |Default licence |                      |
|          |
└──┴──────────────────────┴──────────┴──────┴──────────────────┴──────────────────────┴──────────
──────────────────┴──────────┘
```

TPM - является полным аналогом лицензирования которое использовалось раньше, с токеном и паспортом, без подключения к LM.

или

type=elm

```
/cluster/storage/ds1/licence/list-licence
┌──┬──────────────────────┬──────────┬──────┬──────────────────┬──────────────────────┬──────────
──────────────────┬──────────┐
|Id|Creation date(UTC) |   SSW ID   |Active|  Description   |Comm. Exp.
Date(UTC)|Expiration date(UTC)|Time left|
├──┼──────────────────────┼──────────┼──────┼──────────────────┼──────────────────────┼──────────
──────────────────┼──────────┤
|1 |03.02.2026 16:26:34|ECSS2000009 |*     |ECSS ELM License|03.02.2027 16:26:34
```

```
|14.03.2026 10:49:12 |1d 6h 59m|
|0 |01.01.1990 00:00:00|ECSS DEFAULT|      |Default licence |                    |
|          |
└──┴──────────────────┴────────────┴──────┴────────────────┴────────────────────┴────
_________________┴──────────┘
```

Установка остальных пакетов ecss и их настройка

Выполнить конфигурацию для Sip адаптера , выполнив следующую команду:

на обоих хостах

```
sudo systemctl edit ecss-pa-sip.service
```

Добавить в конфигурационный файл следующие данные:

```
[Service]
LimitNOFILE=65536
```

Перезагрузите конфигурацию

на обоих хостах

```
sudo systemctl daemon-reload
```

Актуализировать конфигурационный файл, выполнив перезагрузку сервера:

```
sudo reboot
```

Для корректной работы кластерной конфигурации необходимо настроить работу **RestFS** на базе **GlusterFS-сервера.**

Следует проверить в файле /etc/dnsmasq.d/ecss-broker, что адреса **primary и secondary broker** соответствуют указанным при инсталляции пакета **ecss-node**.

на обоих хостах

```
cat /etc/dnsmasq.d/ecss-broker
```

Пример содержания файла на **ecss1** и **ecss2** (содержимое файлов должно быть **одинаковым** на обоих серверах):

на обоих хостах

```
address=/primary.broker.ecss/10.0.10.41
address=/secondary.broker.ecss/10.0.10.42
```

glusterfs

Выполните настройку glusterfs для ecss-restfs на **первом хосте (ecss1)**,
для этого **установите** пакеты glusterfs-server и attr **на обоих** хостах:

на обоих хостах

```
sudo apt install -y glusterfs-server attr
```

Запустите glasterd на **обоих** хостах кластера:

на обоих хостах

```
sudo systemctl start glusterd
```

После запуска выполните создание связи с удаленным виртуальным хостом, выполните команду на **ecss1**:

на ecss1

```
sudo gluster peer probe 10.0.10.42
```

Проверьте наличие созданной связи, выполните команду на **ecss2**:

на ecss2

```
sudo gluster peer status
```

Должна появиться информация о первом хосте — ecss1:

на ecss2

```
sudo gluster peer status
Number of Peers: 1

Hostname: ecss1
Uuid: d7319720-ea3a-43a4-a7d5-e97bf9d205b9
State: Peer in Cluster (Connected)
```

Выполните аналогичную проверку на другом хосте — **ecss1**. Вы также должны увидеть информацию о партнере **ecss2** (или его IP-адрес).

на ecss1

```
sudo gluster peer status
Number of Peers: 1

Hostname: 10.0.10.42
Uuid: f405dbb1-4f6d-4f2f-a116-f56197d18788
State: Peer in Cluster (Connected)
```

Создайте кластер для репликации, запустите репликацию и посмотрите ее статус, выполните следующие команды на **ecss1**:

на ecss1

```
sudo gluster volume create ecss_volume replica 2 transport tcp 10.0.10.41:/var/lib/ecss/
glusterfs 10.0.10.42:/var/lib/ecss/glusterfs force
```

на ecss1

```
sudo gluster volume start ecss_volume
```

на обоих хостах

```
sudo gluster volume info
```

Таким образом, статус репликации будет выглядеть следующим образом, необходимо обратить внимание на поля "**Status**" и "**Bricks**" — они должны иметь следующий вид:

```
Volume Name: ecss_volume
Type: Replicate
Volume ID: 3bfc7587-0f85-48ed-9612-21f0d79c6e52
Status: Started
Snapshot Count: 0
Number of Bricks: 1 x 2 = 2
Transport-type: tcp
Bricks:
Brick1: 10.0.10.41:/var/lib/ecss/glusterfs
Brick2: 10.0.10.42:/var/lib/ecss/glusterfs
Options Reconfigured:
transport.address-family: inet
```

```
nfs.disable: on
performance.client-io-threads: off
```

Чтобы смонтировать glusterfs раздел, выполните на **обоих** хостах **ecss1** и **ecss2** следующие действия:

- Создайте новый systemd unit

на обоих хостах

```
sudo nano /etc/systemd/system/ecss-glusterfs-mount.service
```

- Добавьте в него следующую конфигурацию:

```
[Unit]
Description=mount glusterfs
After=network.target
Requires=network.target

[Service]
RemainAfterExit=no
Type=forking
RestartSec=10s
Restart=always
ExecStart=/sbin/mount.glusterfs localhost:/ecss_volume /var/lib/ecss/restfs -o fetch-
attempts=10
ExecStop=/bin/umount /var/lib/ecss/restfs

[Install]
WantedBy=multi-user.target
```

- Добавить unit в автозагрузку на **обоих** хостах.

на обоих хостах

```
sudo systemctl enable glusterd.service
```

на обоих хостах

```
sudo systemctl enable ecss-glusterfs-mount.service
```

- Создайте директорию для корректной установки RestFS на **обоих** хостах:

на обоих хостах

```
sudo mkdir /var/lib/ecss/restfs
```

- Если хосты не могут быть перезагружены, то можно выполнить следующие команды на **обоих** хостах:

на обоих хостах

```
sudo systemctl daemon-reload
```

на обоих хостах

```
sudo systemctl restart ecss-glusterfs-mount.service
```

Проверьте, что раздел подмонтировался, для этого на обоих хостах выполните команду - **df -h**:

на обоих хостах

```
df -h
```

на обоих хостах

```
df -h

/dev/sda8                      4,5G  213M  4,1G   5% /var/log
/dev/sda5                       37G   48M   35G   1% /var/lib/ecss/ecss-media-server/
records
/dev/sda6                       19G   44M   18G   1% /var/lib/ecss/cdr
/dev/sda7                       19G   44M   18G   1% /var/lib/ecss/statistics
/dev/sda9                       19G  7,6G  9,7G  44% /var/log/ecss
localhost:/ecss_volume          46G   59M   44G   1% /var/lib/ecss/restfs
```

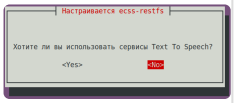
При просмотре информации должен появиться подмонтированный раздел — **localhost:/ecss_volume** 46G 59M 44G 1% **/var/lib/ecss/restfs**

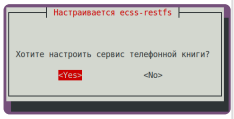
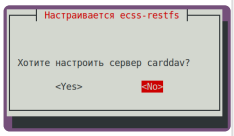
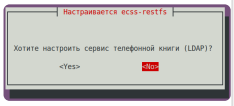
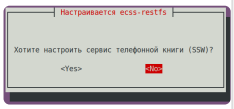
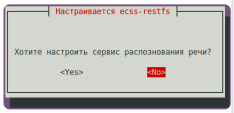
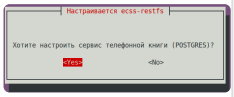
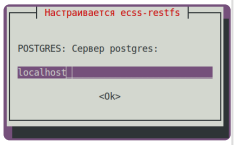
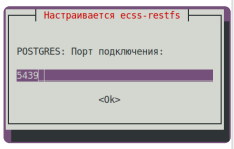
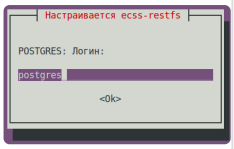
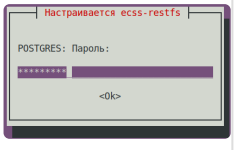
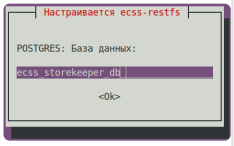
Установка пакета ecss-restfs и его настройка

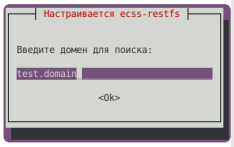
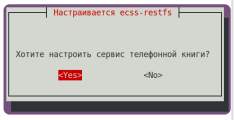
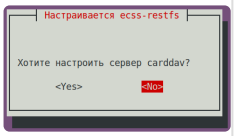
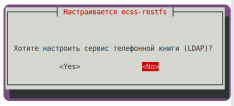
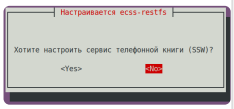
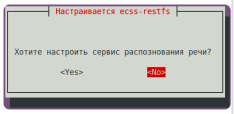
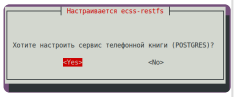
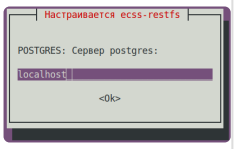
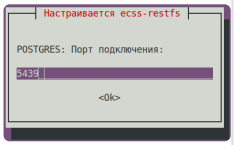
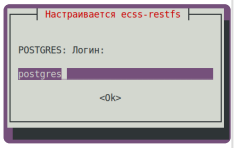
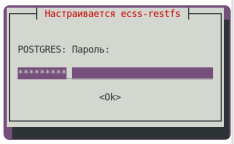
установить на ecss1, затем на ecss2

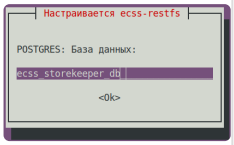
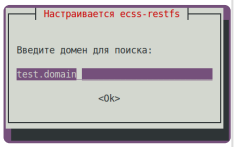
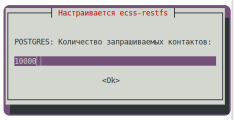
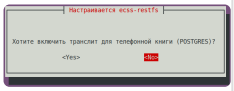
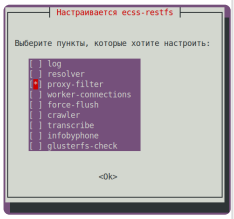
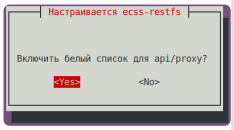
```
sudo apt install -y ecss-restfs
```

Установка **ecss-restfs.** В ходе установки нужно будет ответить на ряд вопросов для создания необходимых конфигурационных файлов. Также инсталлятор предложит установить и настроить пакет Text2speech от Yandex.

Вопросы ecss-restfs	Ответы	Пример
Хотите ли вы использование функции Text To Speeh (Use TTS service)	No (значение по умолчанию)	

Вопросы ecss-restfs	Ответы	Пример
Хотите настроить сервис телефонной книги (Configure phone book)	**Yes** (нужно указать)	
Хотите настроить сервер carddav?	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (LDAP)?	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (SSW)?	**Yes** (нужно указать)	
	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (POSTGRES)?	Yes (значение по умолчанию)	
POSTGRES: Сервер postgres:	localhost (значение по умолчанию)	
POSTGRES: Порт подключения:	5439 (значение по умолчанию)	
POSTGRES: Логин:	postgres (значение по умолчанию)	
POSTGRES: Пароль:	postgres1 (значение по умолчанию)	
POSTGRES: База данных:	ecss_storekeeper_db (значение по умолчанию)	

Вопросы ecss-restfs	Ответы	Пример
Введите домен для поиска:	test.domain (значение по умолчанию)	
POSTGRES: Количество запрашиваемых контактов:	10000 (значение по умолчанию)	
Хотите настроить сервер carddav?	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (LDAP)?	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (SSW)?	No (значение по умолчанию)	
Хотите настроить сервис распознавания речи (Configure speech recognition service)	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (POSTGRES)?	Yes (значение по умолчанию)	
POSTGRES: Сервер postgres:	localhost (значение по умолчанию)	
POSTGRES: Порт подключения:	5439 (значение по умолчанию)	
POSTGRES: Логин:	postgres (значение по умолчанию)	
POSTGRES: Пароль:	postgres1 (значение по умолчанию)	

Вопросы ecss-restfs	Ответы	Пример
POSTGRES: База данных:	ecss_storekeeper_db (значение по умолчанию)	
Введите домен для поиска:	test.domain (значение по умолчанию)	
POSTGRES: Количество запрашиваемых контактов:	10000 (значение по умолчанию)	
Хотите включить транслит для телефонной книги (LDAP)?	No (значение по умолчанию)	
Выберите пункты, которые хотите настроить:	**proxy-filter** (необходимо выбрать)	
Включить белый список для api/proxy?		

белый список адресов прописать в файл /etc/ecss/ecss-restfs/api-proxy.conf

```
# ------------------------------------------------------------
# Формат файла фильтра API (api/proxy):
#   - IP: 192.168.0.1
#   - Домен: example.com
#   - Wildcard: *.wildcard.com
#   - Комментарии: строки, начинающиеся с #
# Пример:
#   192.168.0.1
#   example.com
#   # комментарий
#   *.wildcard.com
# ------------------------------------------------------------
```

ВАЖНО!

В случае успешного проксирование (200) кеш хранится 28 дней, поэтому для применения конфигурации фильтра, в случае изменения, - кеш (/var/cache/ecss/restfs-api) стоит удалить и перезапустить сервис ecss-restfs.

после установки пакета ecss-restfs проверьте наличие wav файлов авто-информатора в директории /var/lib/ecss/restfs/system/sounds/ командой:

на обоих хостах

```
ls -la /var/lib/ecss/restfs/system/sounds/
```

наличие wav файлов в указной выше директории говорит о корректности установки пакета ecss-restfs.

Проверить доступность этих файлов извне, для этой проверки выполним поочередно команду на **ecss1** и **ecss2**:

```
wget http://ecss1:9990/system/sounds/ai_you.wav
```

```
wget http://ecss2:9990/system/sounds/ai_you.wav
```

пример

```
wget http://ecss1:9990/system/sounds/ai_you.wav
--2025-09-16 08:27:36--  http://ecss1:9990/system/sounds/ai_you.wav
Распознаётся ecss1 (ecss1)… 10.0.10.41
Подключение к ecss1 (ecss1)|10.0.10.41|:9990... соединение установлено.
HTTP-запрос отправлен. Ожидание ответа… 200 OK
Длина: 11670 (11K) [audio/x-wav]
Сохранение в: «ai_you.wav»

ai_you.wav.1                                    100%
[=============================================================================
===>]  11,40K  --.-KB/s    за 0s

2025-09-16 08:27:36 (81,5 MB/s) - «ai_you.wav» сохранён [11670/11670]

wget http://ecss2:9990/system/sounds/ai_you.wav
--2025-09-16 08:27:44--  http://ecss2:9990/system/sounds/ai_you.wav
Распознаётся ecss2 (ecss2)… 10.0.10.42
Подключение к ecss2 (ecss2)|10.0.10.42|:9990... соединение установлено.
HTTP-запрос отправлен. Ожидание ответа… 200 OK
Длина: 11670 (11K) [audio/x-wav]
Сохранение в: «ai_you.wav.1»

ai_you.wav.2                                    100%
[=============================================================================
===>]  11,40K  --.-KB/s    за 0s

2025-09-16 08:27:44 (272 MB/s) - «ai_you.wav.1» сохранён [11670/11670]
```

проверочные файлы можно удалить

```
rm ai_you.wav
rm ai_you.wav.1
```

если SSW работает в кластере

В случае кластерной конфигурации, для предотвращения одновременной обработки/конвертации/удаления одного файла (.pcm/.wav/.mp3) на разных серверах ecss1/ecss2 → сервис **ecss-restfs-crawler** должен быть запущен **только на одном из серверов** кластера. Для этого на сервере ecss2 выполнить следующие команды:

на ecss2

```
sudo systemctl disable ecss-restfs-crawler.timer
```

на ecss2

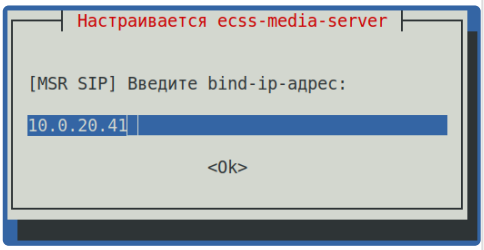
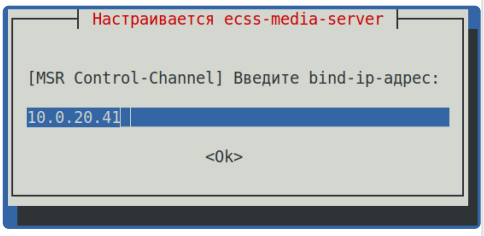
```
sudo systemctl stop ecss-restfs-crawler.timer
```

Установка media и web пакетов ecss и их настройка

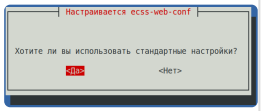
на обоих хостах

```
sudo apt install -y ecss-media-server ecss-web-conf
```

Для медиасервера (*ecss-media-server/MSR*) возможно начальное конфигурирование с записью параметров в файл конфигурации,
для этого нужно провести конфигурацию **transport bind-addr,mcc bind-addres**:

Вопросы ecss-media-server	Ответы для ecss1	Ответы для ecss2	Пример
[MSR SIP] Введите bind-ip адрес (Enter)	**10.0.20.41** (необходимо ввести)	**10.0.20.42** (необходимо ввести)	
[MSR Control-Channel] Введите bind-ip-адрес	**10.0.20.41** (необходимо ввести)	**10.0.20.42** (необходимо ввести)	

Web-конфигуратор позволяет сделать управление системой более наглядным и комфортным. Установка web-конфигуратора не является обязательной, но рекомендуется.
Также при установке пакета **ecss-web-conf** автоматически устанавливается пакет **ecss-subsriber-portal-ui**. Приложение "Портал абонента" системы ECSS-10 позволяет абонентам системы самостоятельно управлять услугами, просматривать информацию по совершенным вызовам, активным конференциям, а также настраивать собственные IVR-скрипты для входящих вызовов. Описание работы веб-конфигуратора приведено в разделе "Портал абонента".

Вопросы ecss-web-conf	Ответы для ecss1/ecss2	Пример
Хотите ли вы использовать стандартные настройки?	Yes (значение по умолчанию)	

После формирования конфигураций по умолчанию переходим в директорию, в которой находятся файлы конфигурации и производим их проверку:

```
cd /etc/ecss/ecss-media-server/
```

```
cat /etc/ecss/ecss-media-server/config.xml
```

Конфигурация для msr: файл - config.xml, в директории conf.d файл default.xml.
По своей сути default.xml — это дополнение config.xml, которое определяет секцию аккаунтов. Это сделано для того, чтобы после обновлений пакета данная конфигурация оставалась неизменной. Вид config.xml представлен здесь: файл конфигурации.

```
<?xml version="1.0" encoding="utf-8"?>
<config date="02:30:33 07.12.2023">
  <general log-level="3" log-rotate="yes" max-calls="8192" max-in-group="512" load-sensor="medi
a" load-delta="10" calls-delta="100" spool-dir-size="100M" log-name="msr.log" log-path="/var/
log/ecss/media-server" use-srtp="disabled" enable-ice-transport="no" ice-update="no"
 aggressive-ice="yes" stun-server="" suspicious-mode="no"/>
  <transport bind-addr="10.0.20.41" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca" user="webrtc@live.com" password="muazkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60" rtcp-timeout="0" rtp-
timeout="350" udp-src-check="no" cn-multiplier="3" port-start="12000" port-range="2048" tias-
in-sdp="no" thread-cnt="2" vid-enc-threads="2" vid-dec-threads="2" video-conf-layout="evenly"
 keyframe-interval="1000" vid-decode-delay="100" silence-threshold="-30" dtmf-flash-disable="no
" video-dscp="0" other-dscp="0" dummy-video-src="/usr/share/ecss-media-server/video/
dummy_video.yuv" video-enc-width="640" video-enc-height="360" finalsilence="1000" rtcp-stat-
dump="yes" silent-codec-switch="yes"/>
  <codec pcma="1" pcmu="2" ilbc="0" gsm="0" g722="3" g726="0" g729="0" speex="0" l16="0" g7221="
0" opus="0" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
  <accounts>
    <!-- <dynamic msr_name="msr.name"
            realm="sip:127.0.0.1:5000"
            dtmf_mode="rfc+inband+info"
            auth_name="user"
            auth_password="password" /> -->
  </accounts>
  <pbyte>
    <mcc bind-addr="10.0.20.41" port="5700"/>
```

```
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

По умолчанию после инсталляции ECSS-10 **активизирует только** кодеки — **pcma, pcmu, g722,** h264, h263-1998, t38. Остальные имеют статус =0 = выкл. **Если требуется** активировать остальные кодеки, в редакторе nano измените приоритет с 0 на цифру приоритета (уровень приоритета не должен повторятся для сервиса audio/video/fax)
sudo nano /etc/ecss/ecss-media-server/config.xml

чтобы активизировать **все** кодеки
строчку -
<codec pcma="1" pcmu="2" ilbc="**0**" gsm="**0**" g722="3" g726="**0**" g729="**0**" speex="**0**" l16="**0**" g7221="**0**" opus="**0**" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
привести к виду -
<codec pcma="1" pcmu="2" ilbc="**6**" gsm="**4**" g722="3" g726="**11**" g729="**5**" speex="**7**" l16="**8**" g7221="**9**" opus="**10**" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>

Недопустимо указывать у разных кодеков одинаковый приоритет, для примера g722="5" g729="5". В этом случае media-server не сможет работать корректно.
За исключением "0" - выключено.

Рассмотрим секцию аккаунтов (файл default.xml):

После инсталляции пакетов в фале будет указан ip-add 127.0.0.1 , и только для одного хоста.

```
cat /etc/ecss/ecss-media-server/conf.d/default.xml
```

```
<?xml version="1.0"?>
<config>
      <accounts>
            <dynamic msr_name="msr.ecss1" realm="sip:127.0.0.1:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>
      </accounts>
</config>
```

Требуется указать в конфигурации для обоих хостов, реальные IP-add.

Необходимо привести содержимое файла к виду указанному снизу (в примере ecss1 имеет IP-add 10.0.20.41/ ecss2 - 10.0.20.42):

на обоих хостах

```
sudo nano /etc/ecss/ecss-media-server/conf.d/default.xml
```

Настройка msr для ecss1 (/etc/ecss/ecss-media-server/conf.d/default.xml)

```
<?xml version="1.0"?>
<config>
  <accounts>
    <dynamic msr_name="msr.ecss1" realm="sip:10.0.20.41:5000" dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password">
      <via dynamic-ifaces="no">
        <iface name="net.20" ip="10.0.20.41"/>
      </via>
    </dynamic>
    <dynamic msr_name="msr.ecss1" realm="sip:10.0.20.42:5000" dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password">
      <via dynamic-ifaces="no">
        <iface name="net.20" ip="10.0.20.42"/>
      </via>
    </dynamic>
  </accounts>
</config>
```

Настройка msr для ecss2 (/etc/ecss/ecss-media-server/conf.d/default.xml)

```
<?xml version="1.0"?>
<config>
  <accounts>
    <dynamic msr_name="msr.ecss2" realm="sip:10.0.20.42:5000" dtmf_mode="rfc+inband+info"
 auth_name="user" auth_password="password">
      <via dynamic-ifaces="no">
        <iface name="net.20" ip="10.0.20.42"/
>
      </via>
    </dynamic>
    <dynamic msr_name="msr.ecss2" realm="sip:10.0.20.41:5000" dtmf_mode="rfc+inband+info"
 auth_name="user" auth_password="password">
      <via dynamic-ifaces="no">
        <iface name="net.20" ip="10.0.20.41"/
>
      </via>
    </dynamic>
  </accounts>
</config>
```

Указываем текущие настройки, согласно которым происходит регистрация msr на core.

Основными параметрами здесь являются: msr_name и realm:

- msr_name — параметр, определяющий название msr. (рекомендуется задавать msr. и к какому хосту он принадлежит, например msr.ecss1);
- realm — определяет адрес для регистрации на ядре. Точка входа по умолчанию — порт 5000.

После изменения конфигурации файлов **/etc/ecss/ecss-media-server/config.xml** и **/etc/ecss/ecss-media-server/conf.d/default.xml** необходимо выполнить перезапуск сервиса **ecss-media-server**, чтобы изменения вступили в силу, следующей командой:

на обоих хостах

```
sudo systemctl restart ecss-media-server
```

Настройка VRRP для SIP-адаптера

Для Astra Linux

Если ECSS-10 установлен на Astra Linux, то необходимо изменить скрипт проверки доступности контрольного SIP-порта **/usr/bin/ecss_pa_sip_port:**

на обоих хостах

```
sudo nano /usr/bin/ecss_pa_sip_port
```

на обоих хостах

```
#!/bin/bash
netstat -plane | grep $1
#nc -z 127.0.0.1 $1
```

Выполните настройку VRRP для SIP-адаптера, для этого на **обоих** хостах создайте файлы следующего содержания:

Глобального конфигурационного файла **keepalived.conf**. На обоих хостах содержимое файла **одинаковое**:

на обоих хостах

```
sudo nano /etc/keepalived/keepalived.conf
```

скопируйте и вставьте следующий текст в файл /etc/keepalived/keepalived.conf

на обоих хостах

```
global_defs {
    vrrp_version 3          # версия протокола VRRP (2 или 3)
    script_user nobody      # ограниченный в правах системный пользователь, от которого будут
запускаться скрипты проверки доступности
    enable_script_security  # не запускать скрипты от root, если часть пути к ним доступна на
запись для обычных пользователей
}

include /etc/keepalived/sip.conf
```

Конфигурационного файла sip.conf:

на обоих хостах

```
sudo nano /etc/keepalived/sip.conf
```

etc/keepalived/sip.conf для ecss1

```
vrrp_script check_sip {
    script "/usr/bin/ecss_pa_sip_port 65535"
    interval 2
    timeout 2
}

# Конфигурация адреса для первого
виртуального адреса SIP-адаптера
vrrp_instance SIP1 {
    state MASTER                    # Исходное
состояние при старте
    interface net.20                # Имя
сетевого интерфейса, на котором будет
работать протокол VRRP
    virtual_router_id 31            #
Уникальный идентификатор роутера (0..255)
    priority 100                    #
Приоритет (0..255) чем выше - тем больше
    advert_int 1                    # Интервал
рассылки уведомлений (с)
    preempt_delay 60                # Интервал
ожидания мастера при старте демона (с) при
исходном состоянии BACKUP

    unicast_src_ip 10.0.20.41     #
Собственный реальный IP-адрес
    unicast_peer {
        10.0.20.42                  # Реальный
 IP-адрес соседа
    }

    virtual_ipaddress {
        # Виртуальный IP-адрес и маска
        # dev - сетевой интерфейс, на
котором будет поднят виртуальный адрес
        # label - метка виртуального
интерфейса (для удобства идентификации)
        10.0.20.31/24 dev net.20 label
net.20:SIP1
    }

    track_script {
        check_sip
    }
}

# Конфигурация адреса для второго
виртуального адреса SIP-адаптера

vrrp_instance SIP2 {
    state BACKUP                        #
Исходное состояние при старте
```

etc/keepalived/sip.conf для ecss2

```
vrrp_script check_sip {
    script "/usr/bin/ecss_pa_sip_port 65535"
    interval 2
    timeout 2
}

# Конфигурация адреса для первого
виртуального адреса SIP-адаптера
vrrp_instance SIP1 {
    state BACKUP                    # Исходное
состояние при старте
    interface net.20                # Имя
сетевого интерфейса, на котором будет
работать протокол VRRP
    virtual_router_id 31            #
Уникальный идентификатор роутера (0..255)
    priority 50                     #
Приоритет (0..255) чем выше - тем больше
    advert_int 1                    # Интервал
рассылки уведомлений (с)
    preempt_delay 60                # Интервал
ожидания мастера при старте демона (с) при
исходном состоянии BACKUP

    unicast_src_ip 10.0.20.42     #
Собственный реальный IP-адрес
    unicast_peer {
        10.0.20.41                  # Реальный
 IP-адрес соседа
    }

    virtual_ipaddress {
        # Виртуальный IP-адрес и маска
        # dev - сетевой интерфейс, на
котором будет поднят виртуальный адрес
        # label - метка виртуального
интерфейса (для удобства идентификации)
        10.0.20.31/24 dev net.20 label
net.20:SIP1
    }

}

# Конфигурация адреса для второго
виртуального адреса SIP-адаптера

vrrp_instance SIP2 {
    state MASTER                        #
Исходное состояние при старте
    interface net.20                    # Имя
сетевого интерфейса, на котором будет
работать протокол VRRP
    virtual_router_id 32                #
Уникальный идентификатор роутера (0..255)
```

etc/keepalived/sip.conf для ecss1

```
    interface net.20                    # Имя
сетевого интерфейса, на котором будет
работать протокол VRRP
    virtual_router_id 32                #
Уникальный идентификатор роутера (0..255)
    priority 50                         #
Приоритет (0..255) чем выше - тем больше
    advert_int 1                        #
Интервал рассылки уведомлений (с)
    preempt_delay 60                    #
Интервал ожидания мастера при старте демона
 (с) при исходном состоянии BACKUP

    unicast_src_ip  10.0.20.41          #
Собственный реальный IP-адрес
    unicast_peer {
         10.0.20.42                     #
Реальный IP-адрес соседа
    }

    virtual_ipaddress {
        # Виртуальный IP-адрес и маска
        # dev - сетевой интерфейс, на
котором будет поднят виртуальный адрес
        # label - метка виртуального
интерфейса (для удобства идентификации)
        10.0.20.32/24 dev net.20 label
net.20:SIP2
    }

}
```

etc/keepalived/sip.conf для ecss2

```
    priority 100                        #
Приоритет (0..255) чем выше - тем больше
    advert_int 1                        #
Интервал рассылки уведомлений (с)
    preempt_delay 60                    #
Интервал ожидания мастера при старте демона
 (с) при исходном состоянии BACKUP

    unicast_src_ip  10.0.20.42          #
Собственный реальный IP-адрес
    unicast_peer {
         10.0.20.41                     #
Реальный IP-адрес соседа
    }

    virtual_ipaddress {
        # Виртуальный IP-адрес и маска
        # dev - сетевой интерфейс, на
котором будет поднят виртуальный адрес
        # label - метка виртуального
интерфейса (для удобства идентификации)
        10.0.20.32/24 dev net.20 label
net.20:SIP2
    }

    track_script {
        check_sip
    }
}
```

В данном случае была добавлена поддержка виртуальных интерфейсов со связью Master - Backup. Для ***ecss1*** основным будет являться ***net.20:SIP1,*** а запасным — ***net.20:SIP2*** соответственно. Важно учесть, что в конфигурации учтено использование адресных переменных. В секции interface требуется указать, на каком интерфейсе будут слушаться VRRP-сообщения, а в секции virtual_ipaddress требуется указать интерфейс, на котором будет восстановлен виртуальный адрес.

Выполните перезапуск keepalived:

на обоих хостах

```
sudo systemctl restart keepalived.service
```

для проверки правильности установки выполните команду ifconfig, в конце выходного сообщения вы должны видеть новый интерфейс - net.20:SIP1/ net.20:SIP2 (ecss1/ecss2) с выбранным IP-адресом

на обоих хостах

```
sudo ifconfig
```

```
sudo ifconfig

. . .
net.20:SIP1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 10.0.20.31 netmask 255.255.255.0 broadcast 0.0.0.0
ether 08:00:27:6d:4f:02 txqueuelen 1000 (Ethernet)
```

Дальнейшая настройка

snpmd

Настройка SNMP описана в разделе "Управление и мониторинг по протоколу SNMP".

Настройка безопасности. SSH

Выполните настройку сервера SSH:

на обоих хостах

```
sudo nano /etc/ssh/sshd_config
```

Укажите в конфигурационном файле порт и адрес, по которому можно обратиться к серверу:

Настройка ssh для ecss1 (/etc/ssh/sshd_config)	Настройка ssh для ecss2 (/etc/ssh/sshd_config)
# This is the sshd server system-wide configuration file. See # sshd_config(5) for more information. # This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin # The strategy used for options in the default sshd_config shipped with # OpenSSH is to specify options with their default value where # possible, but leave them commented. Uncommented options override the # default value. **Port 2000** #AddressFamily any **ListenAddress 10.0.10.41** #ListenAddress :: <...>	# This is the sshd server system-wide configuration file. See # sshd_config(5) for more information. # This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin # The strategy used for options in the default sshd_config shipped with # OpenSSH is to specify options with their default value where # possible, but leave them commented. Uncommented options override the # default value. **Port 2000** #AddressFamily any **ListenAddress 10.0.10.42** #ListenAddress :: <...>

Перезапустите ssh:

на обоих хостах

```
sudo systemctl restart ssh.service
```

Перезагрузка обоих хостов

на обоих хостах

```
sudo reboot
```

Добавление услуг

Установка всех услуг указных в лицензии:

выполняется в CoCon

```
/cluster/storage/ds1/ss/install ds1@ecss1 ss_*
```

Результат:

```
/cluster/storage/ds1/ss/install ds1@ecss1 ss_*
Successfully installed: /var/lib/ecss/ss/ss_dnd.xml
Successfully installed: /var/lib/ecss/ss/ss_clir.xml
...
Successfully installed: /var/lib/ecss/ss/ss_chunt.xml
```

10.6 Первоначальная конфигурация системы с помощью web-конфигуратора

пример первоначальной конфигурации через Web интерефейс

10.6.1 Исходные данные

- Установка системы завершена;
- Система готова к дальнейшей настройке;
- Интерфейсы запущены.

Рекомендуется использовать последние доступные версии браузеров. Рекомендуемые к использованию браузеры: Яндекс, Chrome.

Для начала конфигурации системы необходимо зайти в web-конфигуратор по адресу http://<Ipadd SSW>.

Для определения и регистрации в системе планируются:

- Абоненты с номерами 42001, 42002 , 42003, 42004 ,42005, 42006, 42007, 42008, 42009, 42010;
- Транк в сторону шлюза.

10.6.2 Подготовка к работе

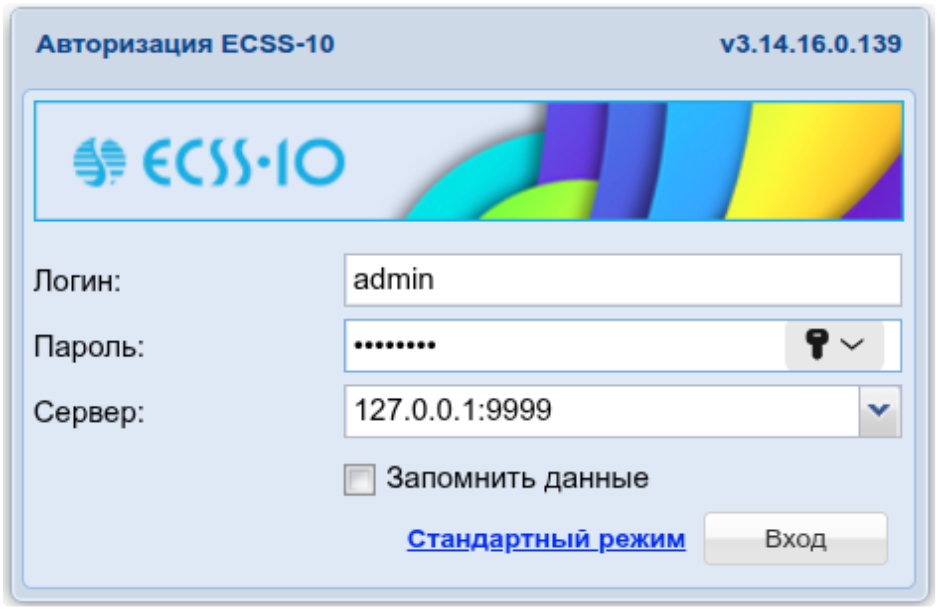


Рисунок 3 — Вход в web-конфигуратор (окно авторизации)

В окне авторизации необходимо ввести значения, определенные при установке web-конфигуратора.

> ⚠ Стандартные значения для авторизации:
> **Логин:** admin
> **Пароль:** password

После входа в систему будет видна основная рабочая область с иконками приложений, а также строка состояния с доступными опциями, в частности:

- 1 — выход из системы;
- 2 — выбор домена;
- 3 — выбор языка.

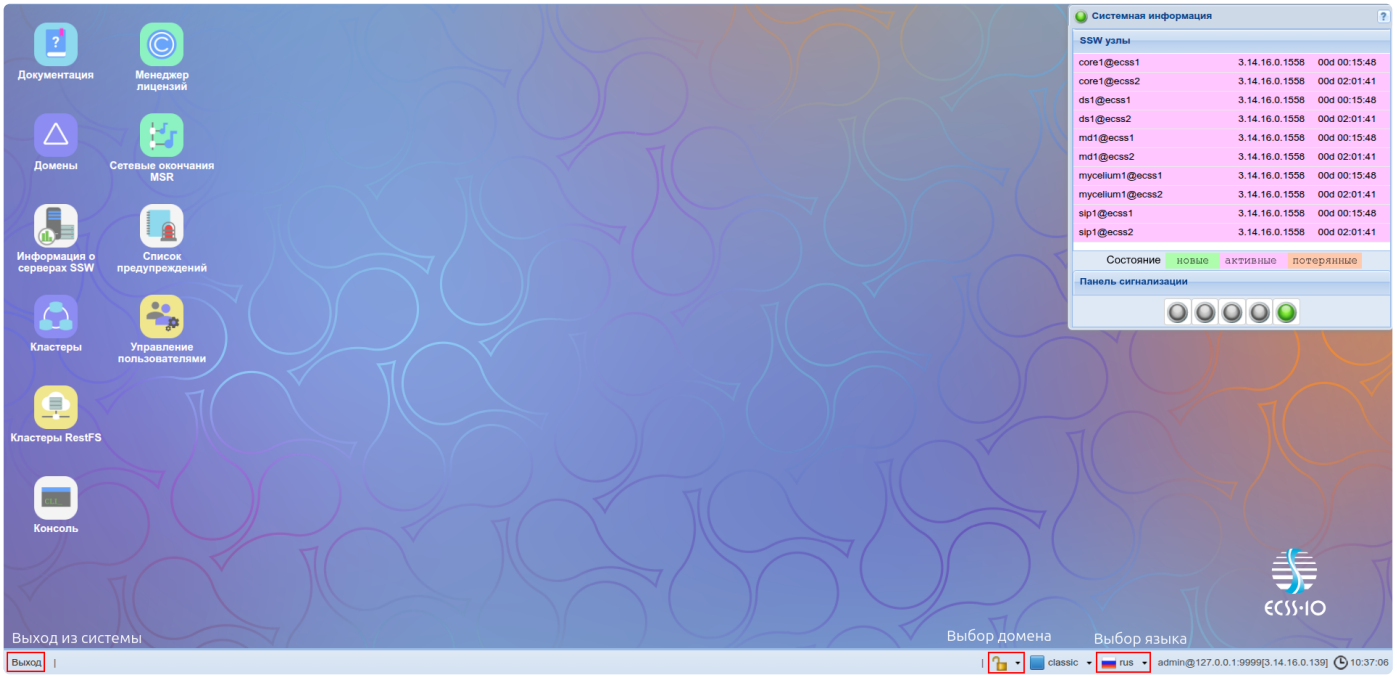


Рисунок 4 — Вид рабочей области web-конфигуратора

10.6.3 Создание учетной записи оператора

После авторизации для повышения безопасности при эксплуатации программного коммутатора рекомендуется создание учетных записей для операторов, а также сменить пароль для пользователя **admin**.

Для создания новой учетной записи оператора используется приложение "Управление пользователями" ("User manager"):

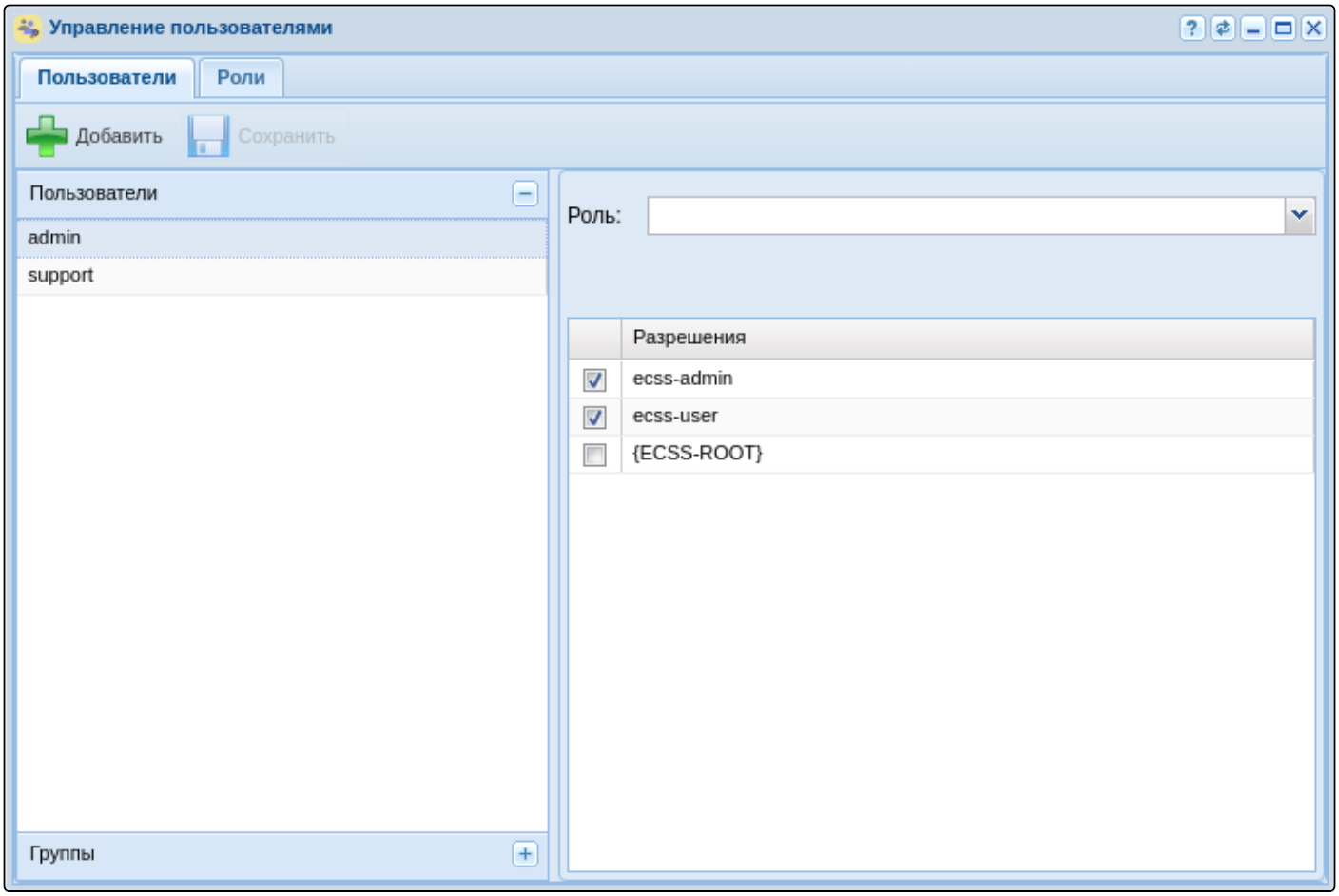


Рисунок 5 — Вид приложения "*Управление пользователями*"

Нажмите кнопку "*Добавить*"  . В открывшемся окне требуется определить новую учетную запись, для этого:

1. В поле "*Имя*" введите логин учетной записи, например "test";
2. В поле "*Пароль*" и "*Подтверждение*" введите пароль для пользователя, например "testpassword";
3. Определите уровень прав доступа для пользователя через выбор текущих разрешений или с помощью ролей, например **ecss-user**. Описание можно посмотреть в разделе "Управление пользователями".

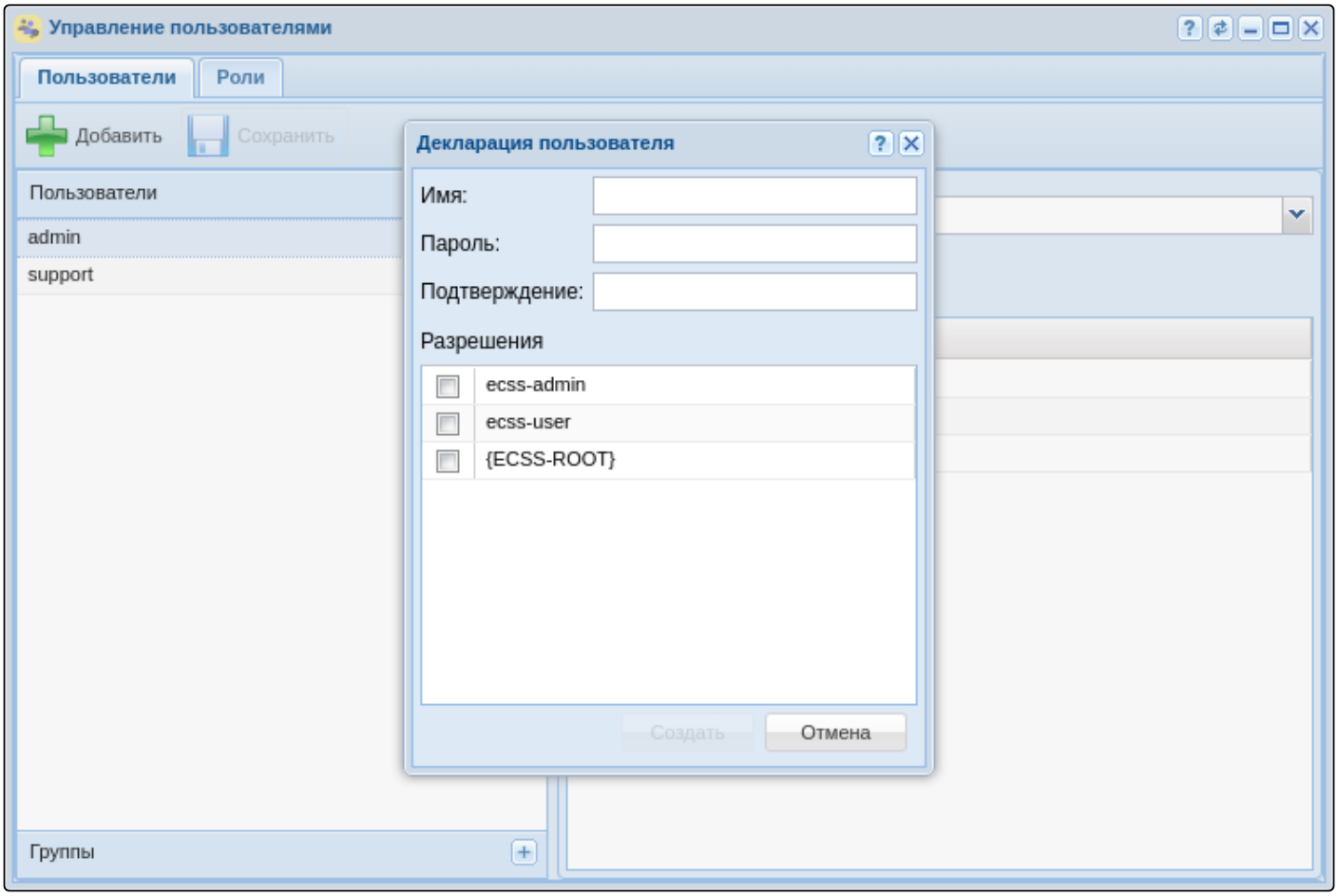


Рисунок 6 – Диалоговое окно создания учетной записи оператора

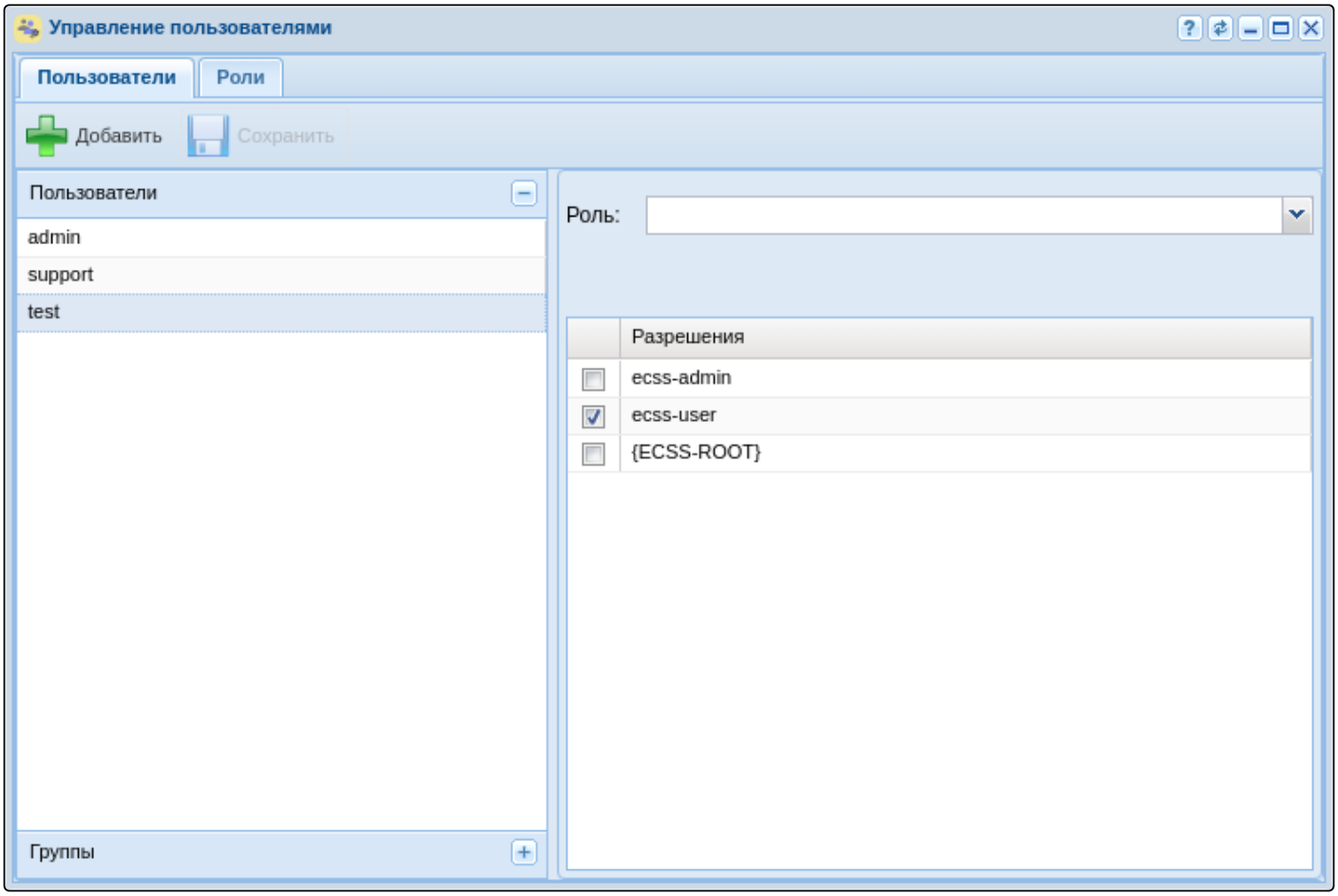


Рисунок 7 — Вид приложения с созданной учетной записью оператора

Для смены пароля рядом с названием пользователя нажмите на кнопку редактировать. В появившемся диалоговом окне введите:

1. Старый пароль (для пользователя "**admin**" по умолчанию паролем является "**password**");
2. Новый пароль;
3. Подтвердите новый пароль.

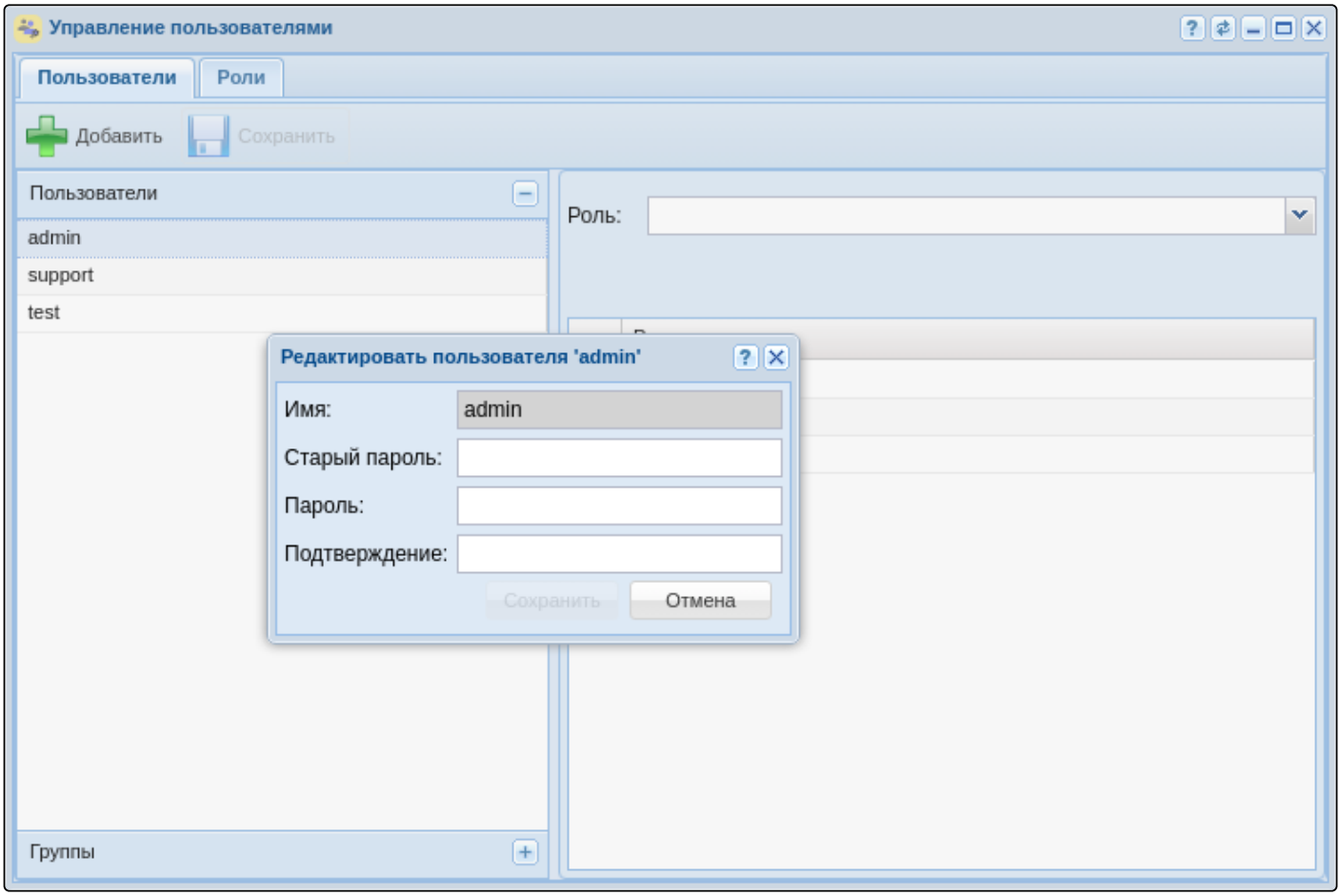
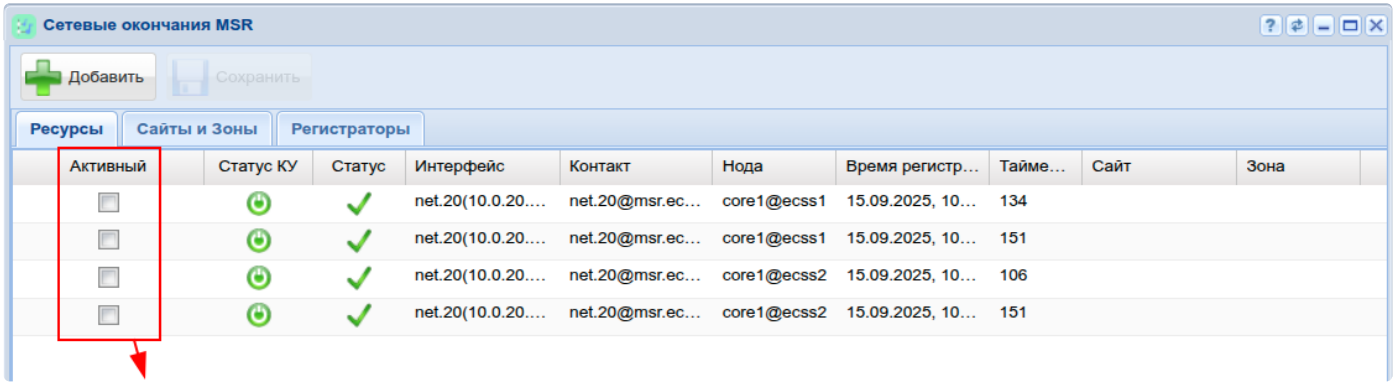


Рисунок 8 — Диалоговое окно редактирования учетной записи

10.6.4 Добавление медиа-ресурсов.

Для того чтобы начать работу с медиа-сервером сразу после его регистрации, необходимо активировать сетевые интерфейсы MSR. Приложение web-конфигуратора "Сетевые окончания MSR" ("MSR registrars") служит для настройки медиа-ресурсов.



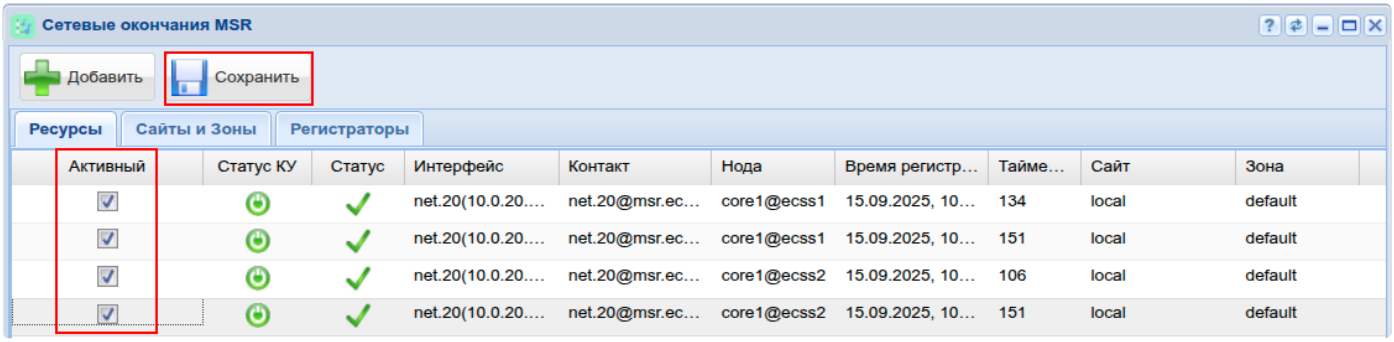
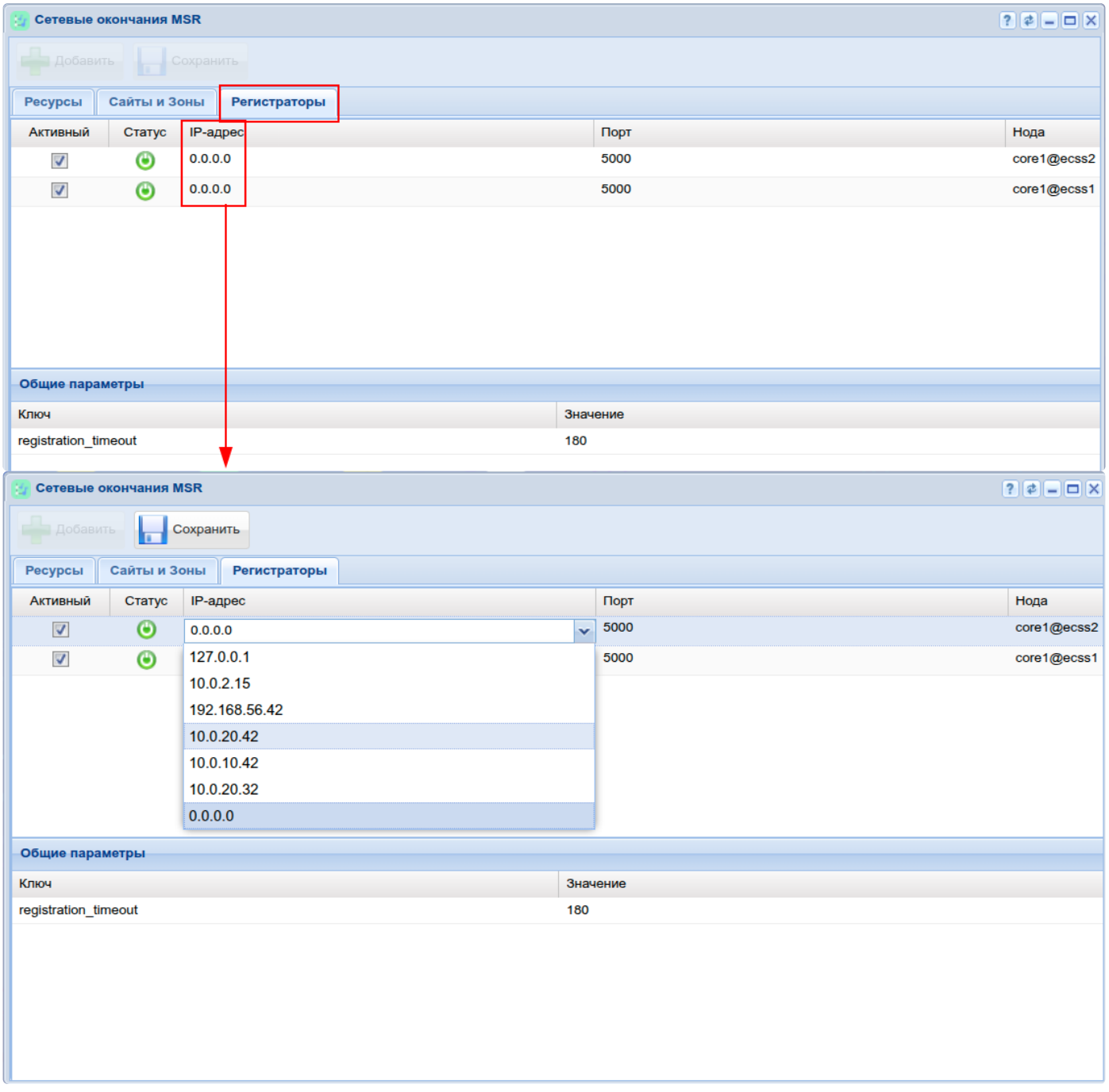


Рисунок 9 — Сетевые окончания MSR

Установить ipадрес для приема соединений от медиасервера, изменив значение 0.0.0.0 на реальные адреса MSR:



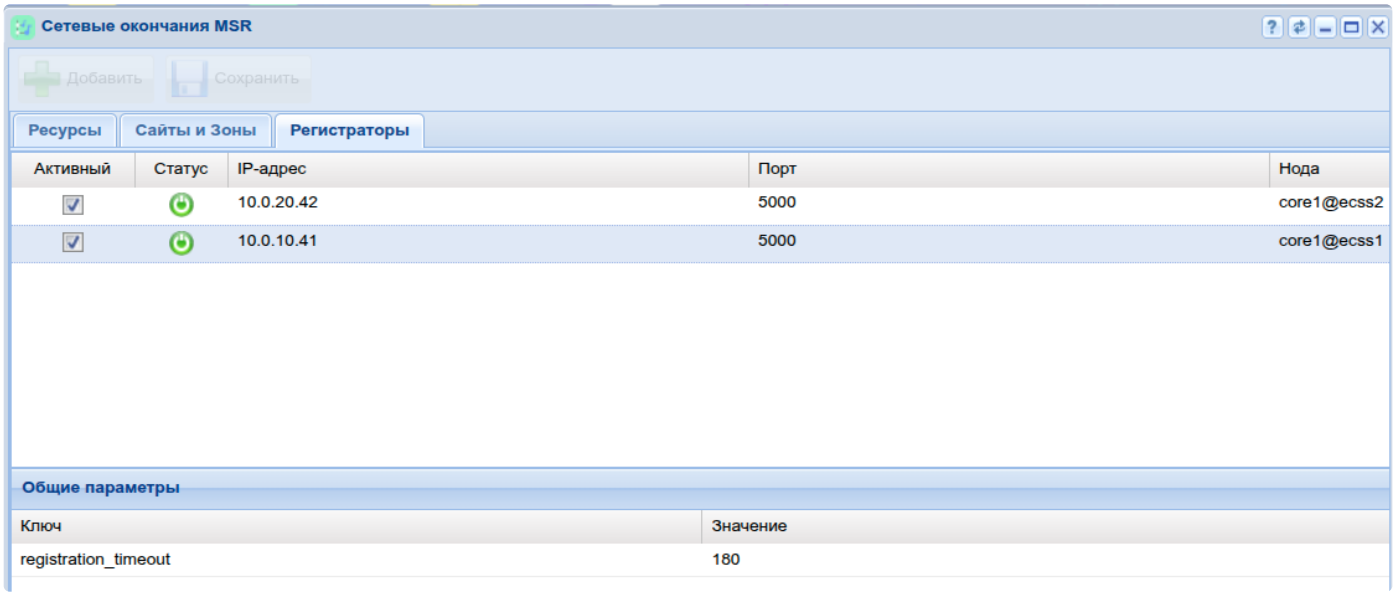


Рисунок 10 — Установка IP-add регистратора MSR

10.6.5 Создание IP-set (sip-транспорта) и присвоение его к домену

IP-set — совокупность одного или нескольких IP-адресов из одной подсети и списка слушающих UDP/TCP-портов, которые будут открыты SIP-адаптером на данных адресах.

Для настройки интерфейса необходимо открыть приложение "Кластеры ("Clusters").

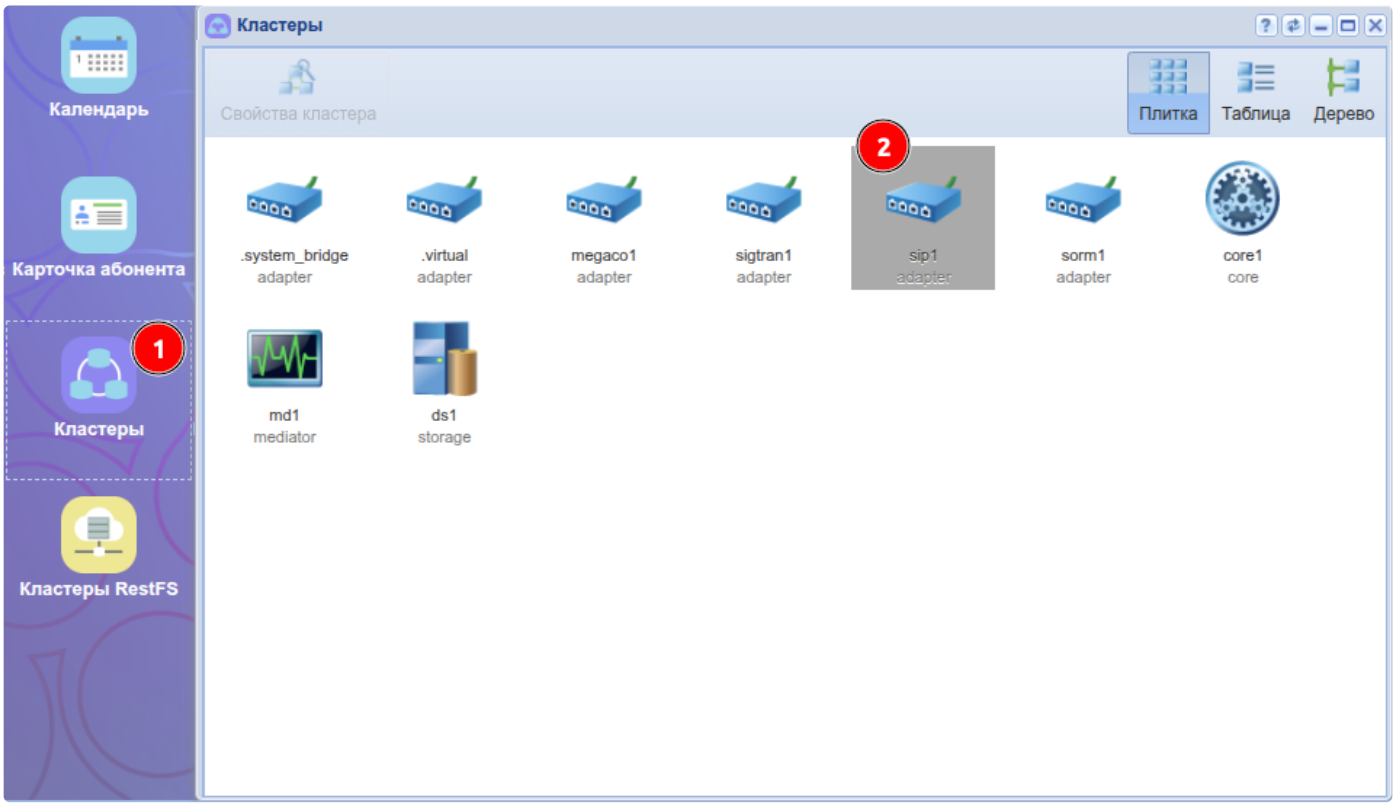


Рисунок 11 — Вид приложения "Кластеры"

Для создания новой группы IP-адресов (IP-set) выделите кластер SIP-адаптера "sip1" и нажмите на кнопку *"Свойства кластера"* (или нажмите на иконку кластера двойным щелчком левой кнопки мыши).

В появившемся диалоговом окне перейдите на вкладку *"Транспорт"*. Далее нажмите кнопку *"Добавить"*. Появится новая группа, для редактирования полей двойным щелчком кнопки мыши нажмите на нужное:

1. Переименуйте группу адресов (IP-set), например "test_set";

2. Задайте порт, по которому будет происходить обращение к домену, например 5060;

3. Разверните вновь созданную группу нажатием на треугольник слева от названия группы;

4. Определите адрес для ноды SIP-адаптера, согласно примеру конфигурации. Для системы без резервирования укажите 10.0.20.31:

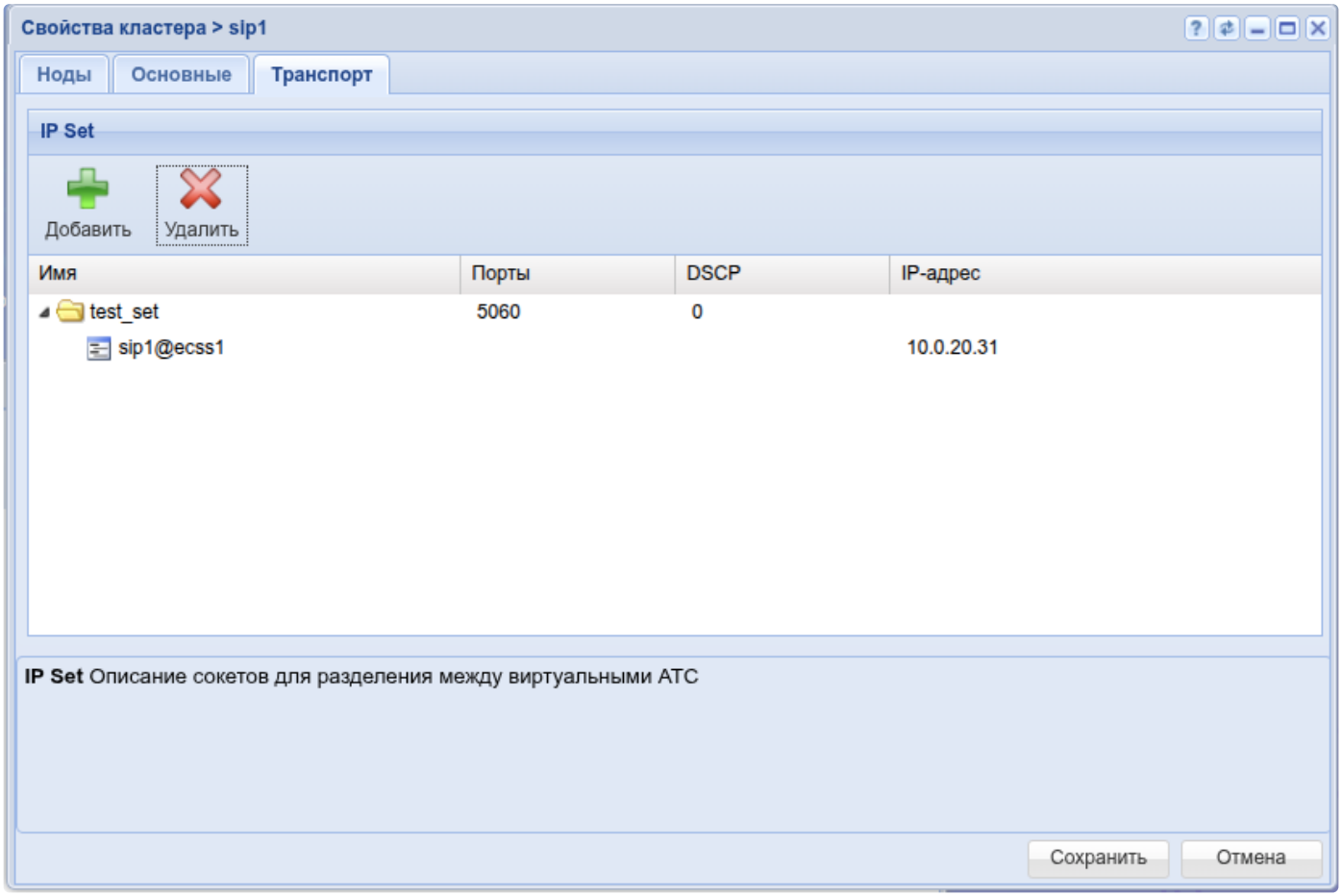


Рисунок 12 — Вид приложения "Кластеры/Транспорт"

Для системы с резервированием укажите 10.0.20.31 и 10.0.20.32:

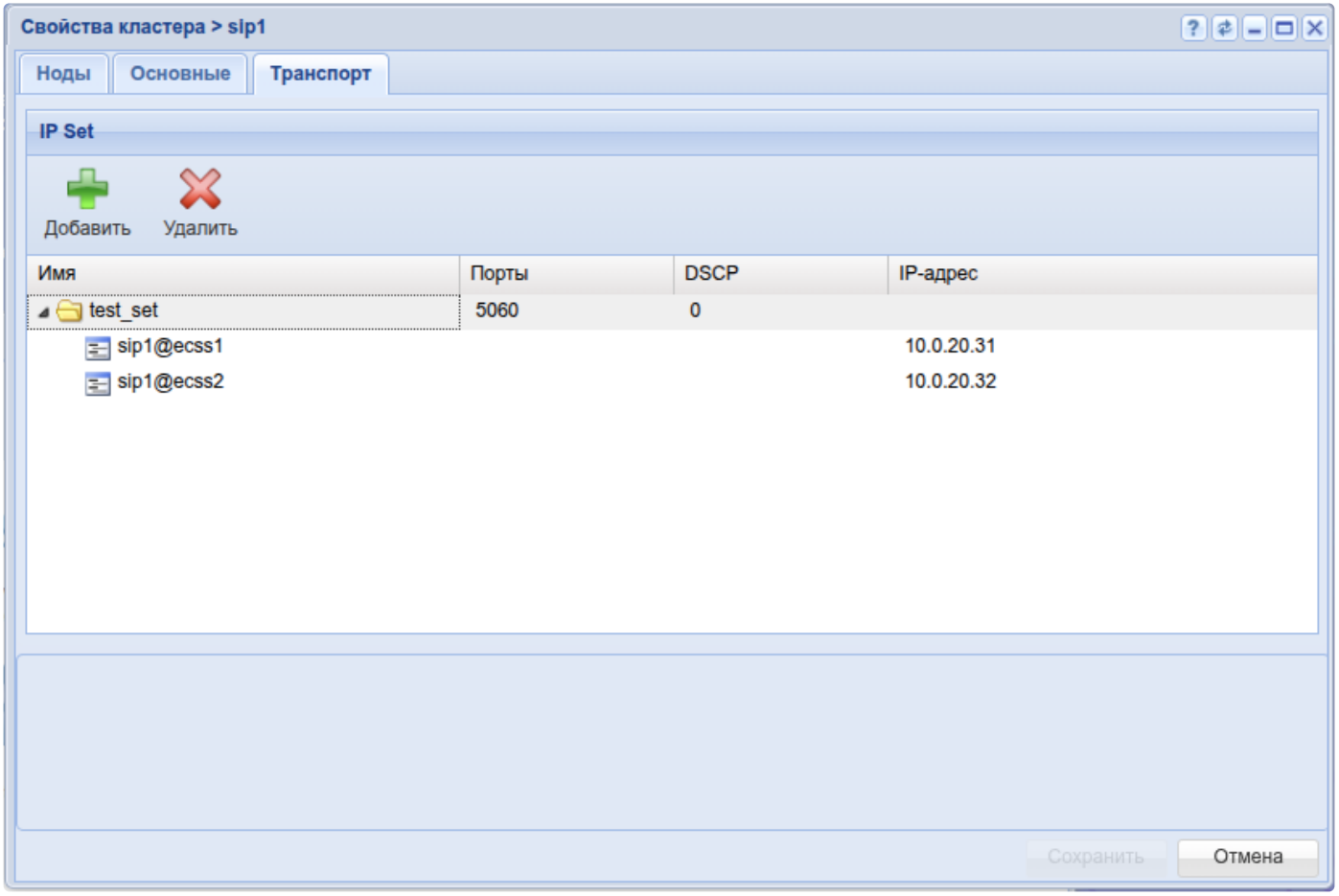


Рисунок 13 — Вид приложения "Кластеры/Транспорт" для кластера

Нажмите кнопку *"Сохранить"* для применения настроек.

10.6.6 Создание домена

Домен - Виртуальная PBX внутри нашего SSW. Таких виртуальных PBX может быть создано необходимое для проекта кол-во.

Для создания домена необходимо войти в приложение *"Домены" ("Domain")*. В открывшемся окне требуется создать домен, для этого:

1. Нажмите кнопку "*Добавить домен*":

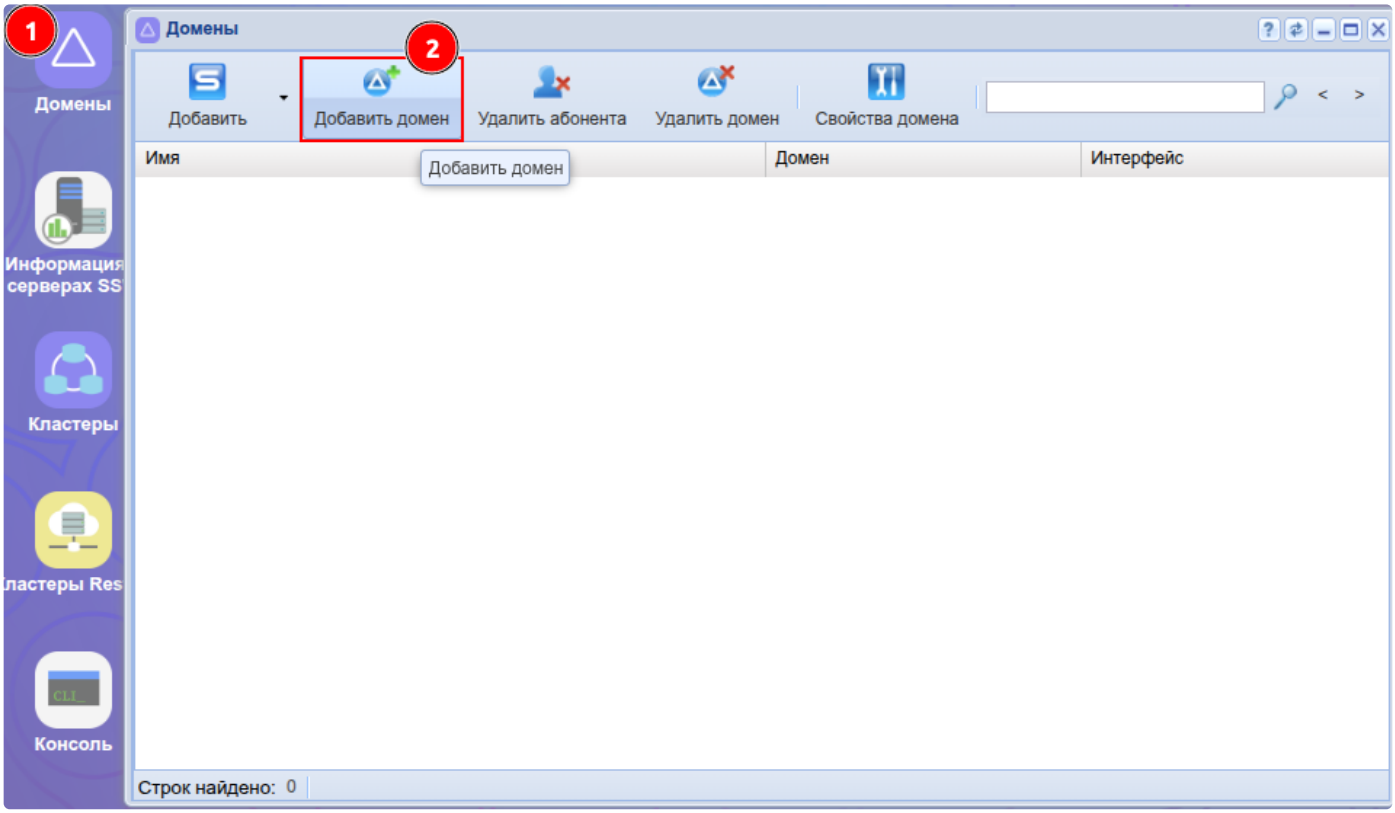


Рисунок 14 — Добавление домена в систему

2. В появившемся диалоговом окне доступны следующие настройки:

- *Имя (Name)* — индивидуальное имя виртуальной АТС;
- *Профиль услуг (SS profile)* — системный профиль дополнительных услуг. Этот профиль будет скопирован с тем же именем во вновь создаваемый домен и для всех услуг из этого профиля будет автоматически разрешён доступ через access-list;
- *IVR профиль (IVR profile)* — профиль IVR, заданный в приложении "Редактор IVR ограничений".

Введите название домена, например "test_domain";

3. Нажмите кнопку "*Ок*":

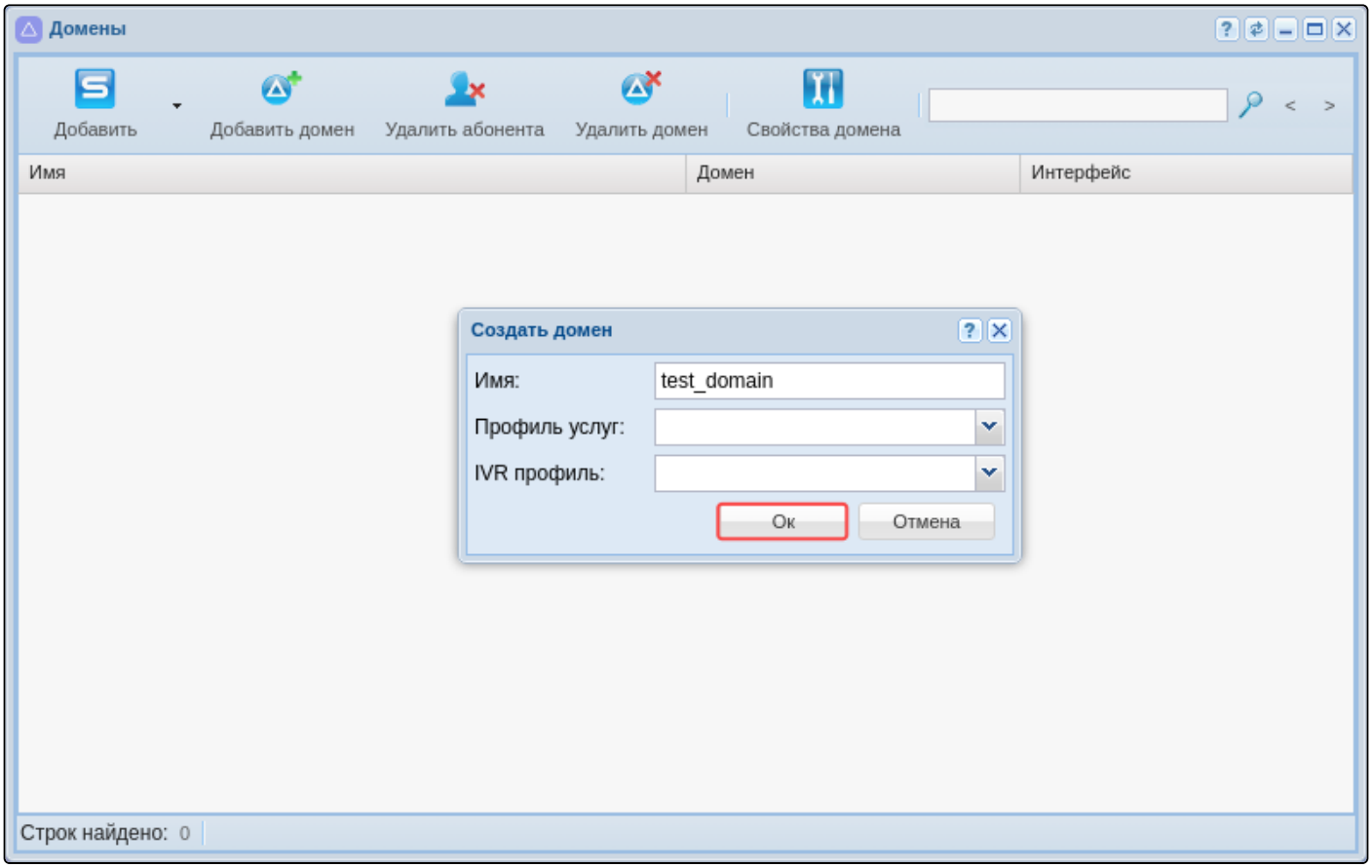


Рисунок 15 – Определение настроек нового домена

4. Нажмите кнопку *"Обновить"* в приложении.

Созданный домен будет отображаться в текущей конфигурации:

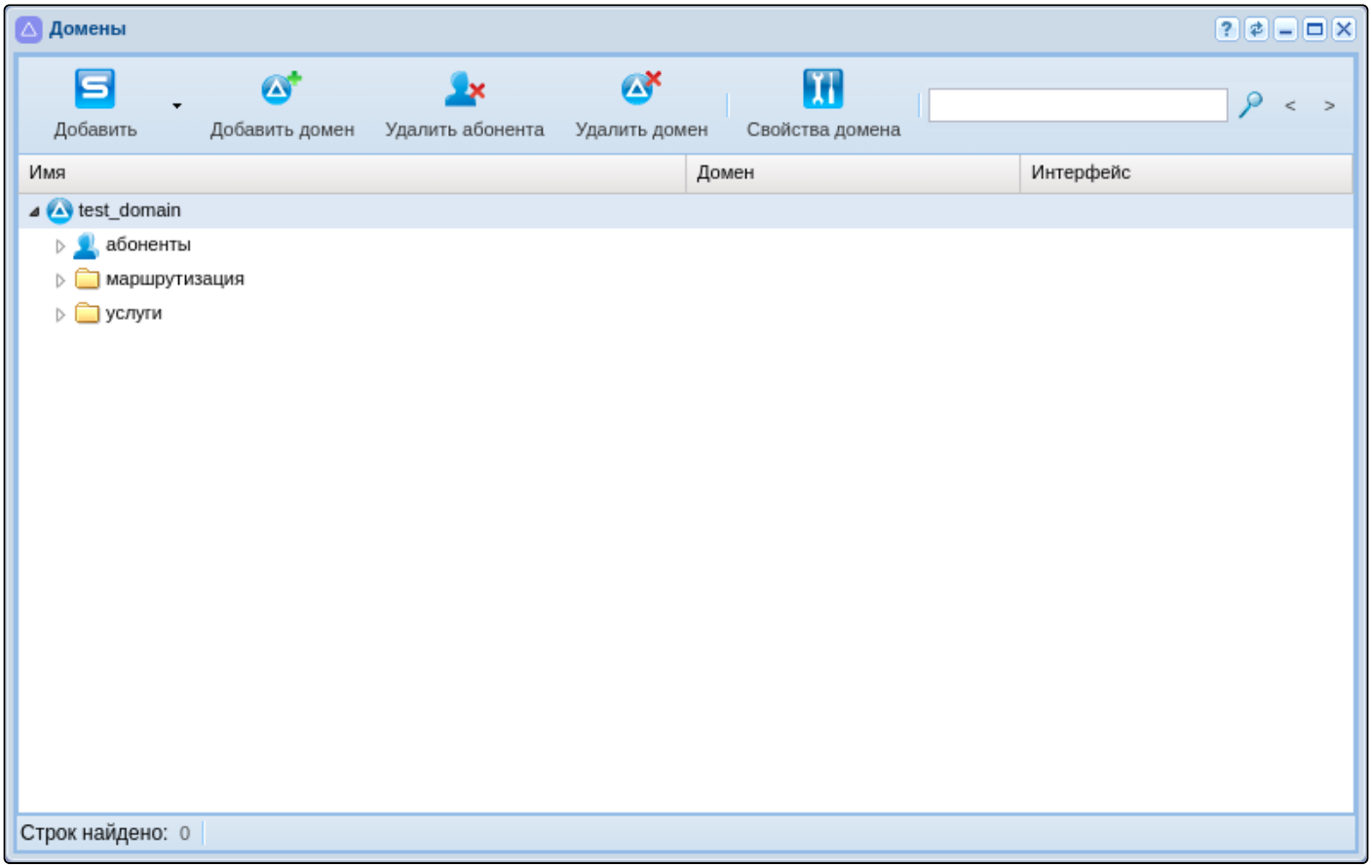


Рисунок 16 — Отображение созданного домена

Для редактирования текущего домена его необходимо выбрать в системе. Для перехода в домен воспользуйтесь опцией выбора домена (см. пункт 2 на рисунке "Вид рабочей области web-конфигуратора").

После выбора домена, согласно текущей конфигурации системы, станут доступны все приложения:

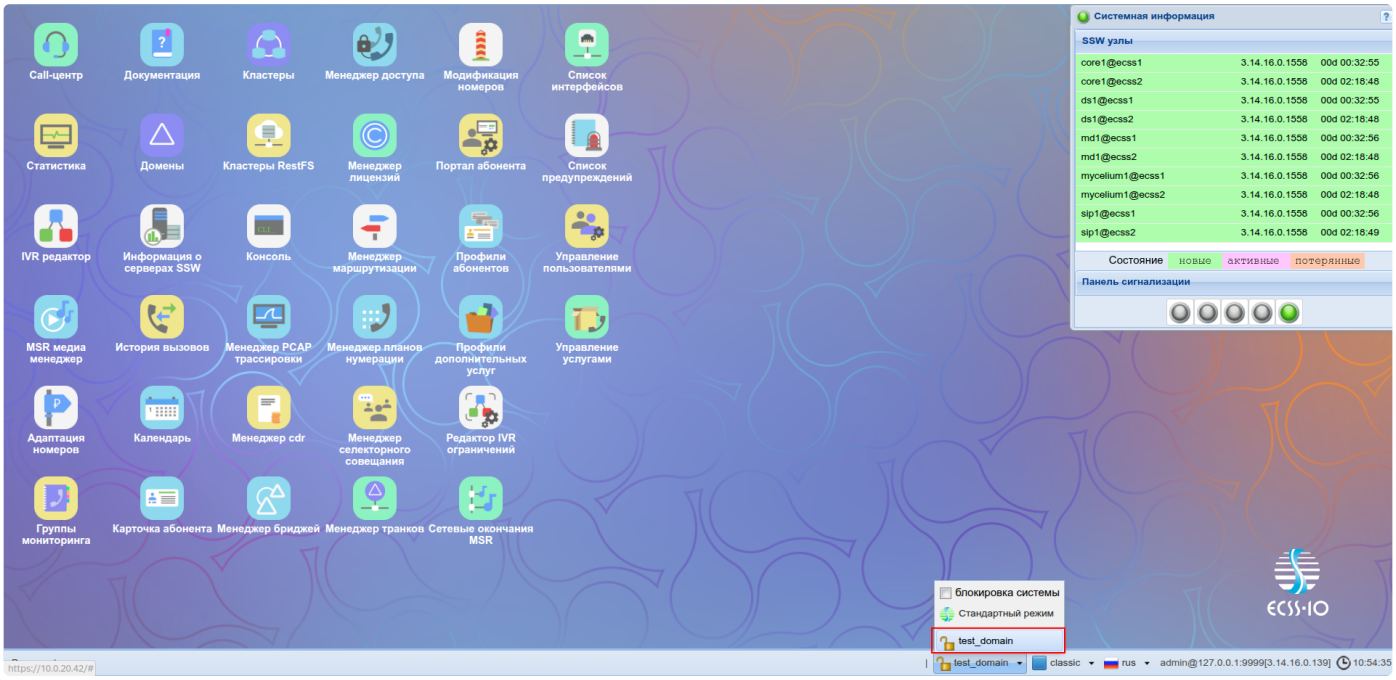


Рисунок 17 — Отображение приложений в текущей конфигурации системы

Для того чтобы **привязать группу адресов к домену**, необходимо вернуться в приложение *"Домены" ("Domain")*, выделить домен и перейти в настройки нажатием на кнопку *"Свойства домена"* или двойным нажатием левой кнопки мыши на домен.

В настройках списка откройте ветку "SIP", в ней "SIP транспорт", далее выберите в поле IP set созданную группу адресов. Нажмите кнопку *"Сохранить"* для применения настроек.

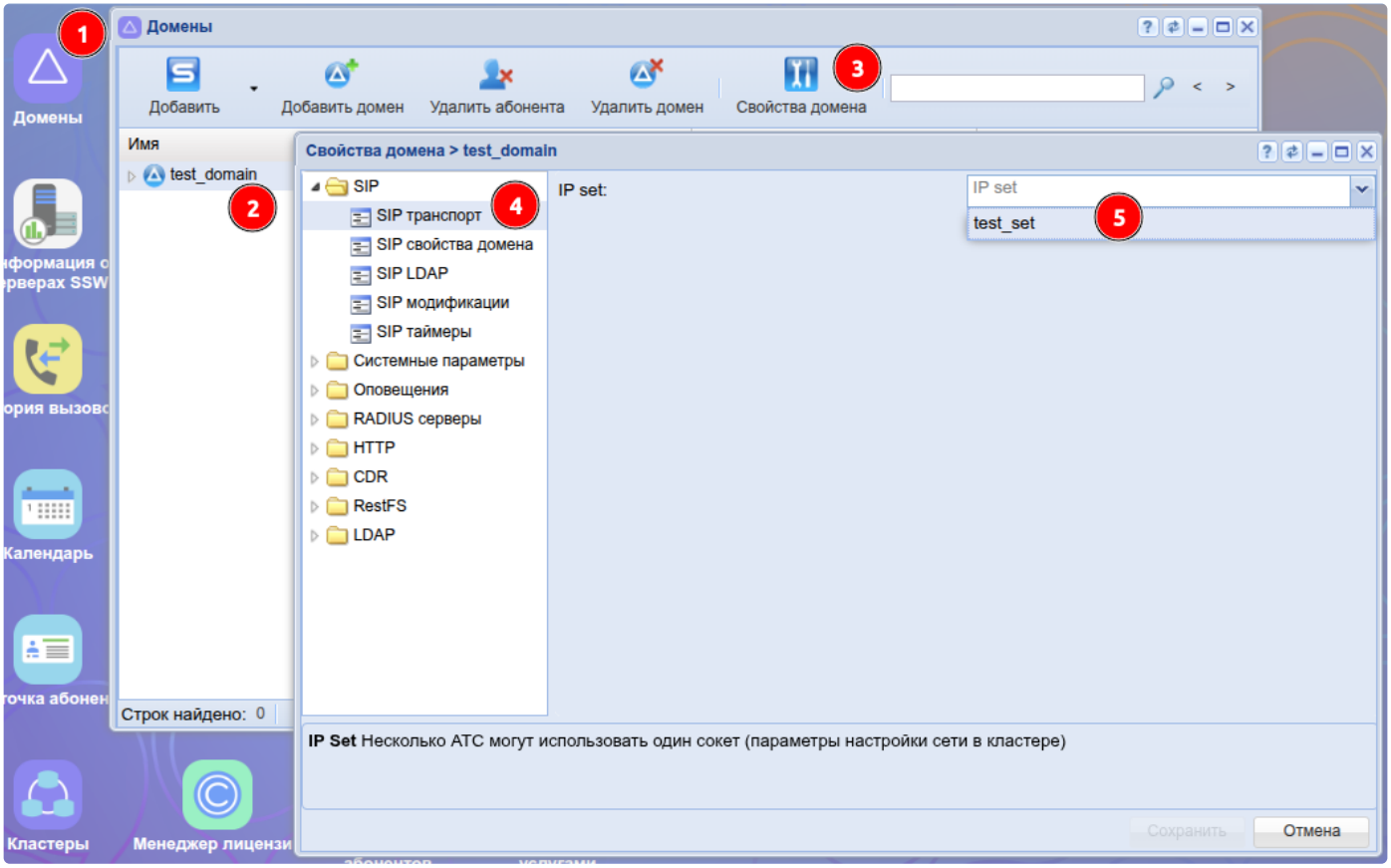


Рисунок 18 — Окно настройки для SIP транспорта

Выбрать требуемые услуги для созданного домена, которые будут использоваться (для примера для абонентского домена транковые услуги не востребованы , и наоборот для транзитного домена не нужно указывать услуги для абонента).

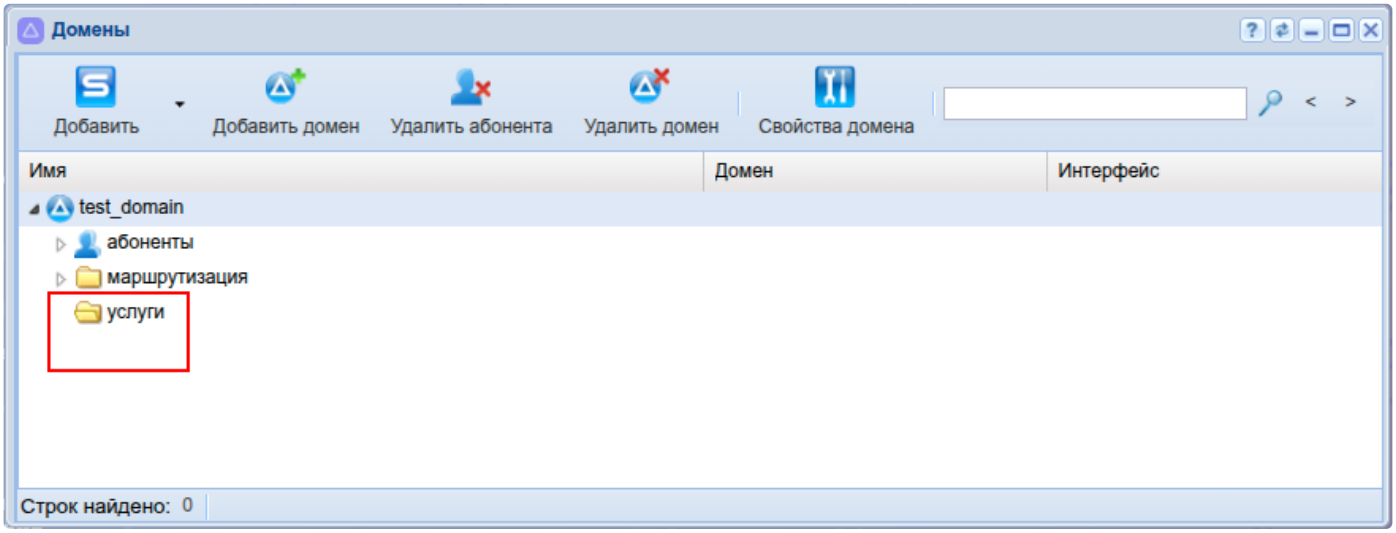


Рисунок 19 — После создания нового домена, услуги отсутствуют.

1. Выбрать приложение "Управление услугами"
2. Выбрать колонку "Установленные услуги" (услуги указанные в лицензии для SSW)

3. Выбрать необходимые для указанного домена услуги
4. Сохранить конфигурацию

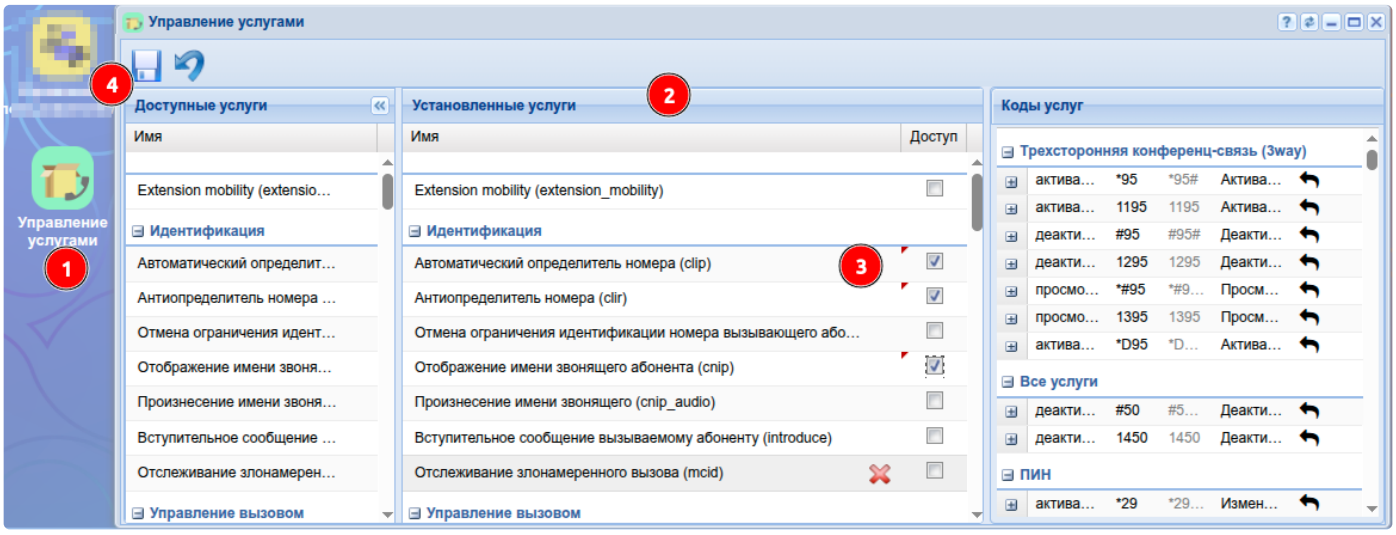


Рисунок 20 — Выбор услуг для абонентов / транков нового домена.

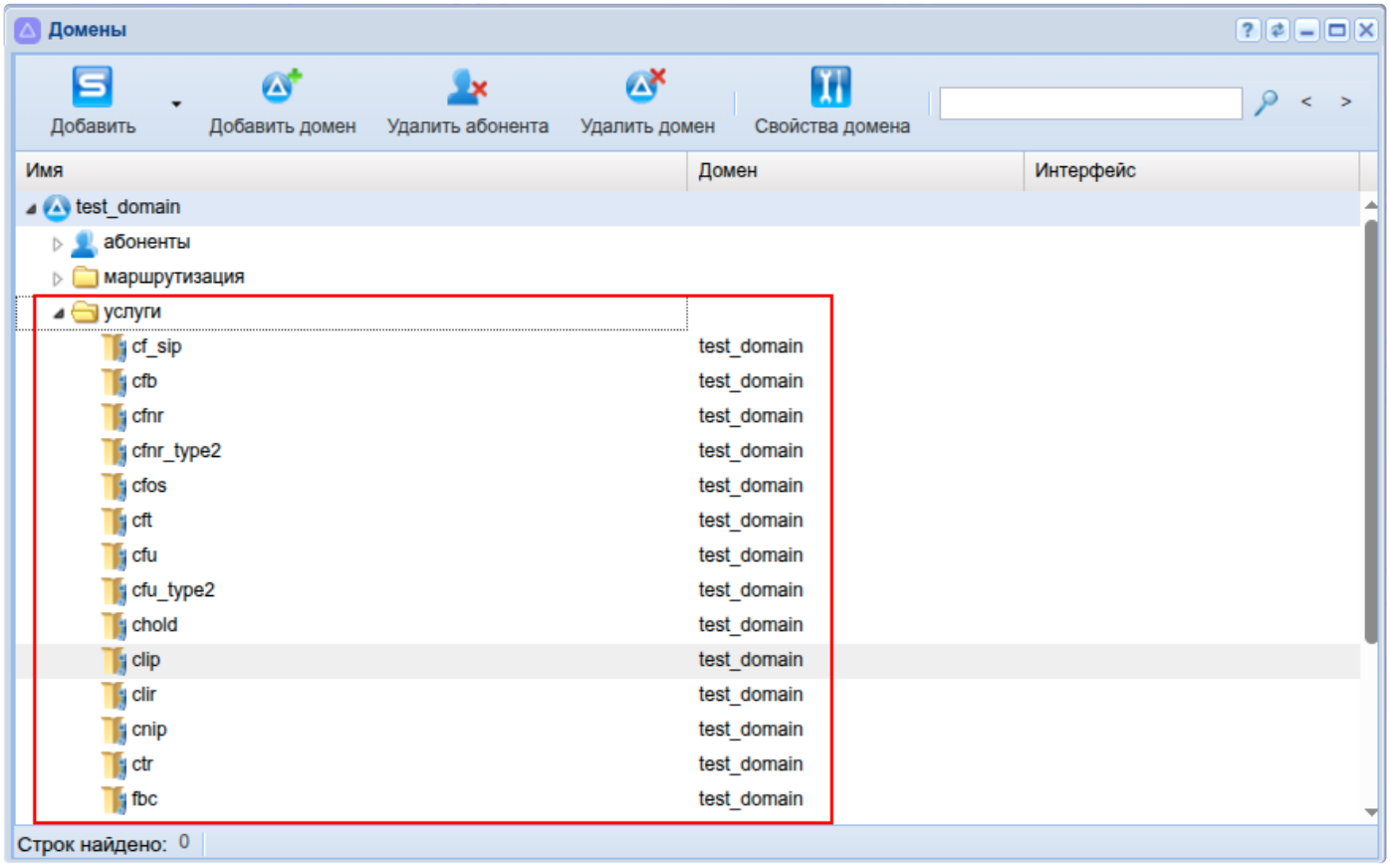


Рисунок 21 — Список услуг теперь доступных для абонентов нового домена.

10.6.7 Создание абонентов

Для создания и редактирования параметров абонентов в системе используется приложение "Карточка абонента" ("Subscriber card").

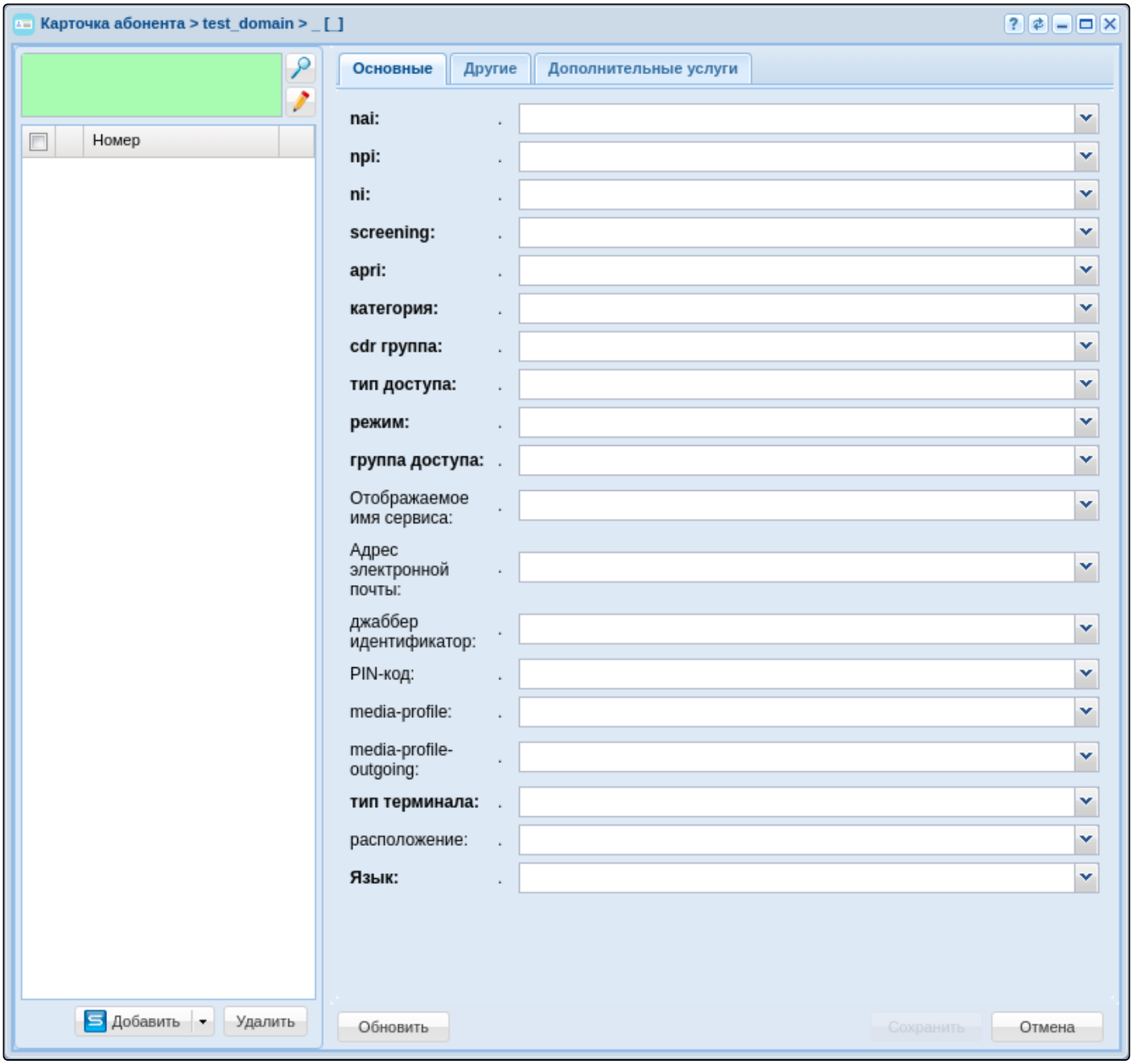


Рисунок 22 — Вид приложения "Карточка абонента"

В конфигурации возможно создание SIP-абонентов и виртуальных абонентов.

Для пользователей, имеющих физическое окончание, используется функционал SIP-абонента, в то время как виртуальный абонент используется тогда, когда нужна функциональность без физических окончаний. Например, номер для обращения к ivr-скрипту.

Для создания новых абонентов нужно нажать на кнопку *"Добавить"*.

В появившемся диалоговом окне укажите следующие параметры:

- *Имя интерфейса* — номер или группа номеров, который присваивается абоненту, строка, например {42001-42010} или abcd;
- *Контекст* — контекст маршрутизации, выбираем тот, который создали, например "test_name", или по умолчанию - "default_routing";
- Группа интерфейсов — имя группы объединяющей нескольких абонентов, для примера "sip";
- *Алиас как пользователь* — настройка, которая связывает номер сущности, алиас и пользователя одним и тем же названием, в примере настройка активирована;

- *Авторизация* — порядок проверки подлинности прав пользователя на доступ к данным, в примере используется"register";
- *Логин* — использую настройку номер как логин для простоты;
- *Пароль* — можно задать или использовать сгенерированный системой.

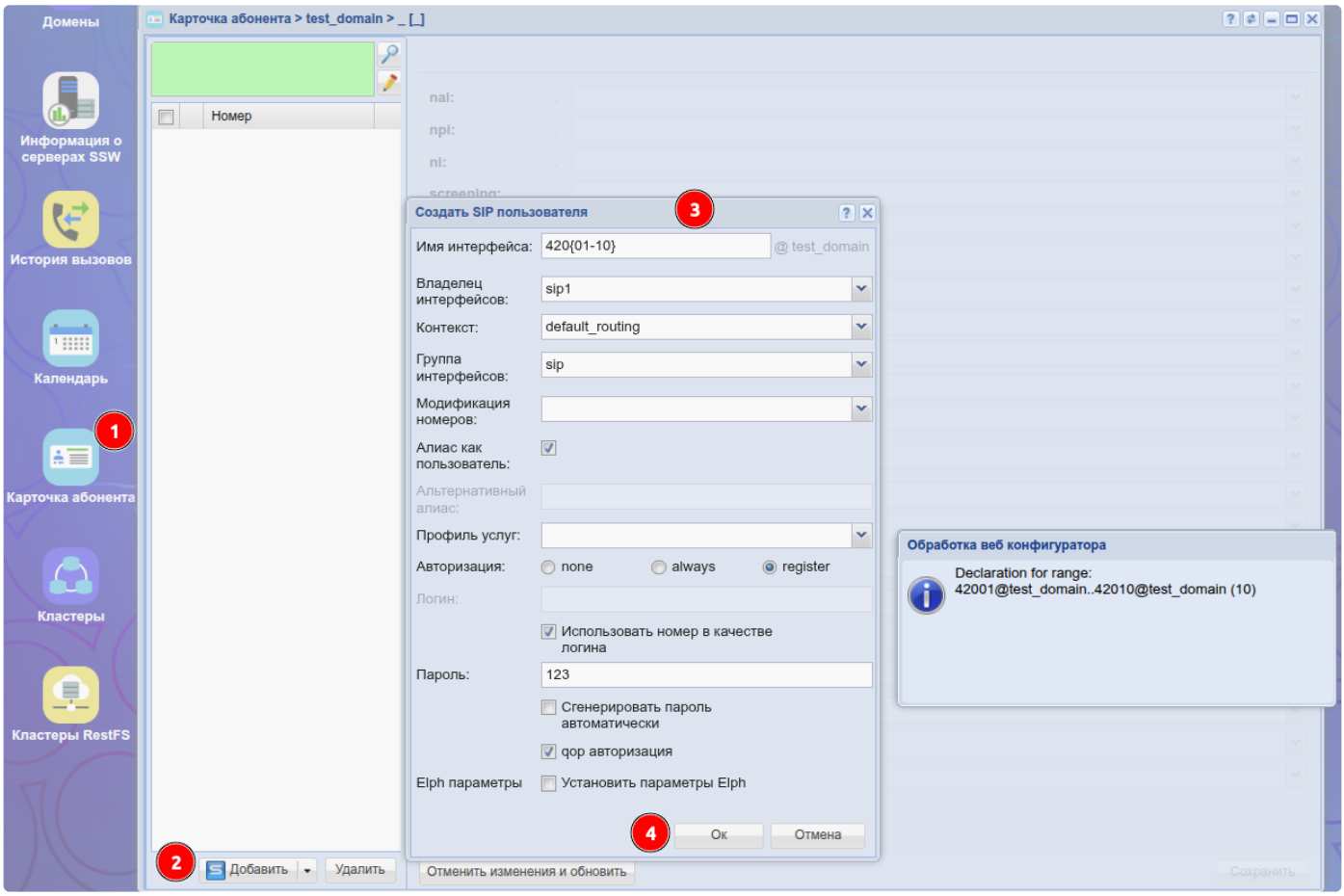


Рисунок 23 — Пример создания 10 абонентов в домене "test_domain"

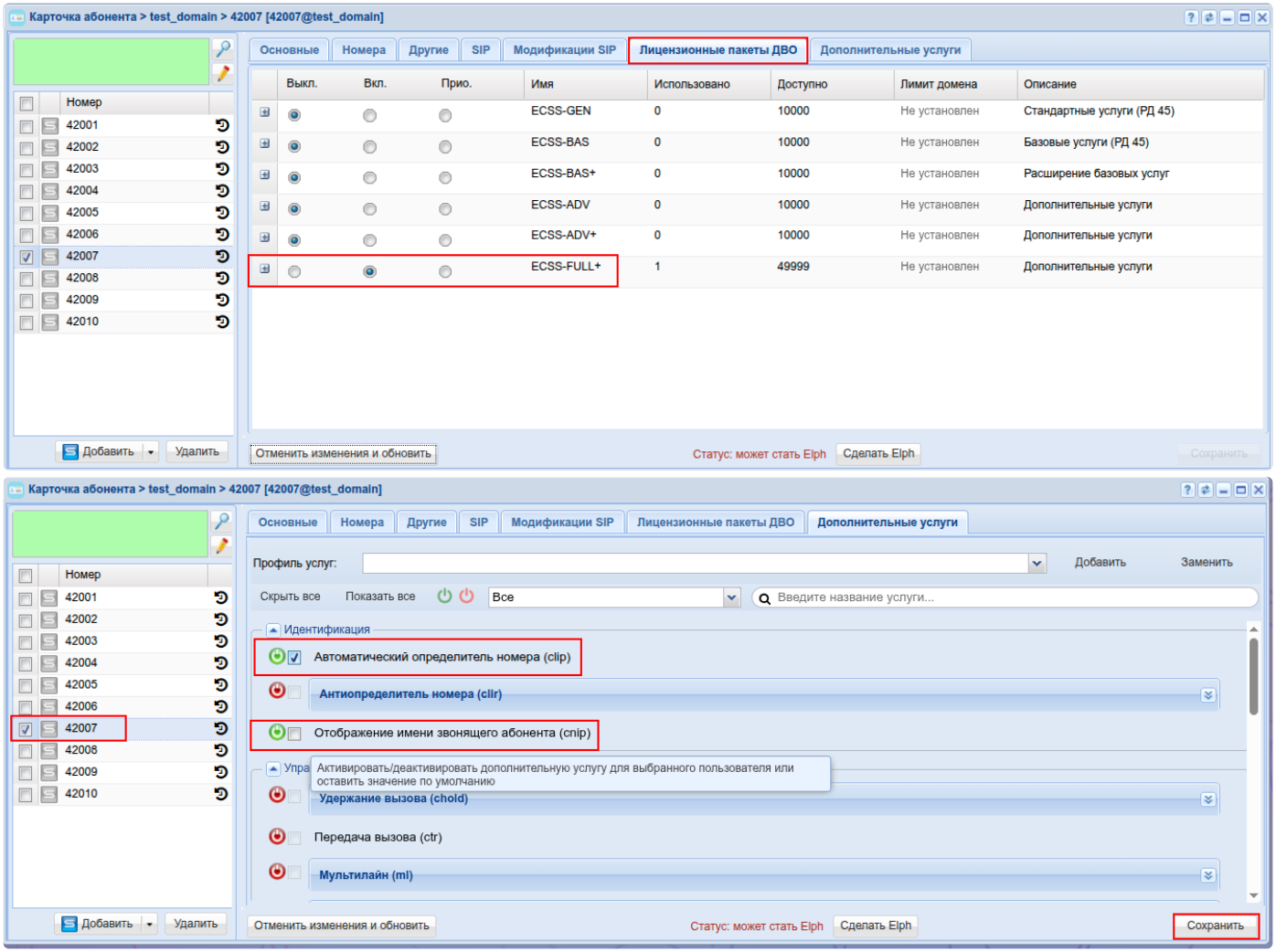


Рисунок 24 – Пример добавления услуг для абонента

10.6.8 Создание и применение контекстов маршрутизации для домена

Маршрутизация отвечает за поиск номера и последующую адресацию вызова. Для корректной работы системы должен быть настроен как минимум один контекст маршрутизации.

Настройка маршрутизации осуществляется в приложении "Менеджер маршрутизации" ("Routing manager") .

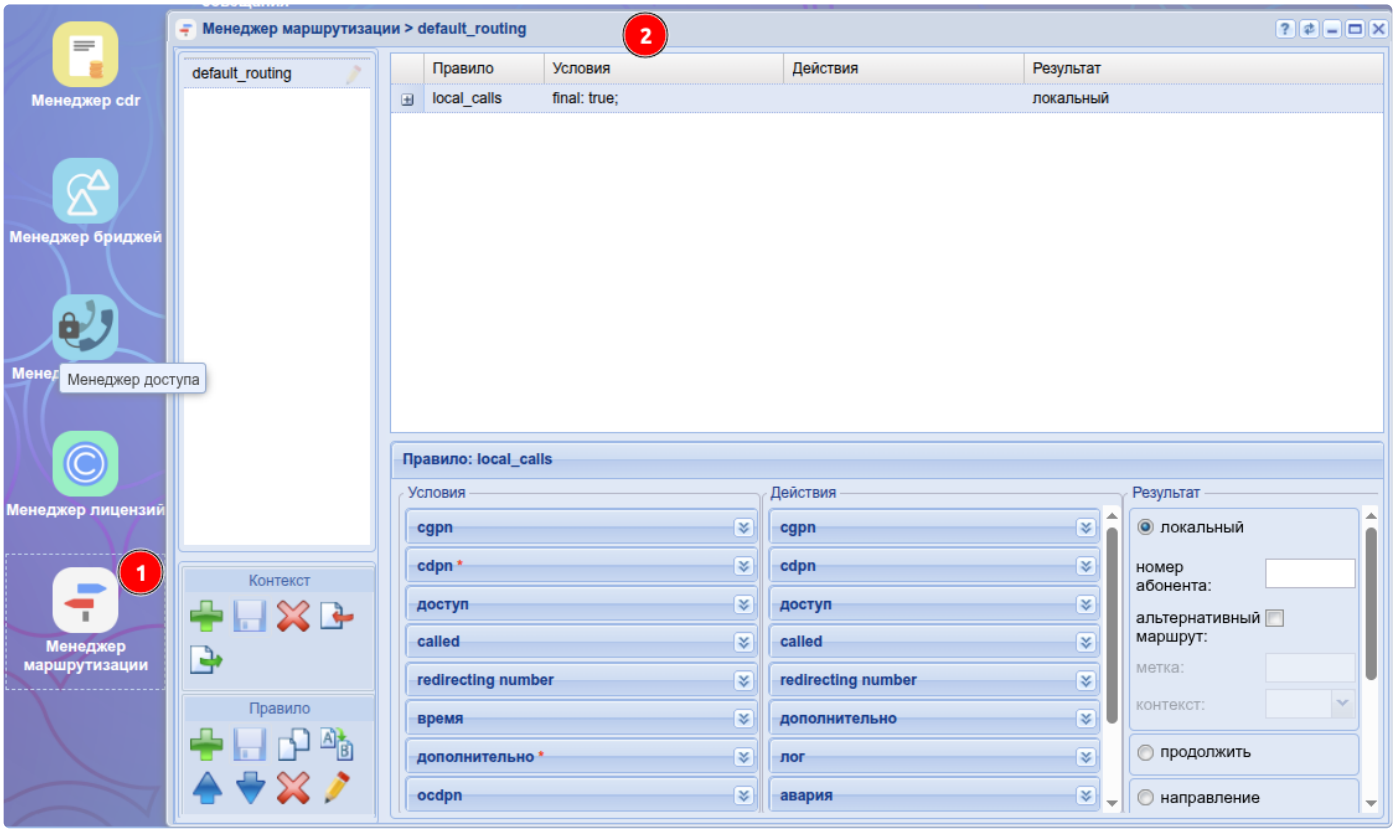


Рисунок 25 — Вид приложения "Менеджер маршрутизации"

В качестве примера создадим контекст, а затем несколько правил в нём, для этого:

1. В левой части окна в разделе "Контекст" нажмите на кнопку *"Создать контекст"*  ;
2. В открывшемся диалоговом окне обозначьте имя контекста:

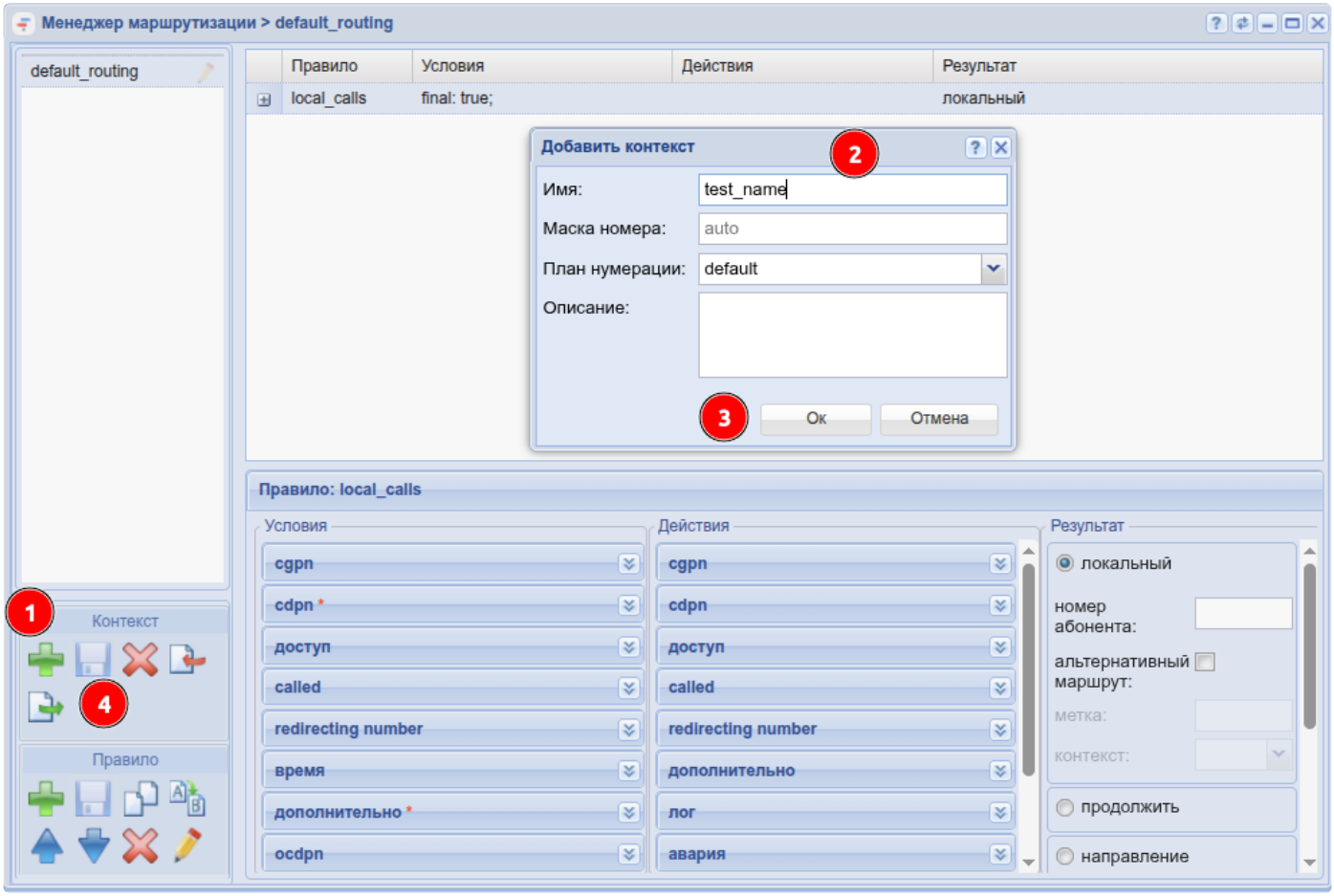


Рисунок 26 — Вид приложения "Менеджер маршрутизации" добавление нового контекста маршрутизации

3. Нажмите *"Сохранить контекст"*  ;

Создадим 4 правила в данном контексте:

- rule1 — правило для выхода на транк TAU-72;
- rule2 — правило локальной маршрутизации для номеров 42001-42005;
- rule3 — правило для выхода в ivr;
- rule4 — правило-исключение.

Для создания нового правила нужно выделить созданный контекст и нажать кнопку *"Создать правило"*. В появившемся окне нужно ввести название правила. Затем сохранить вновь созданные правила.

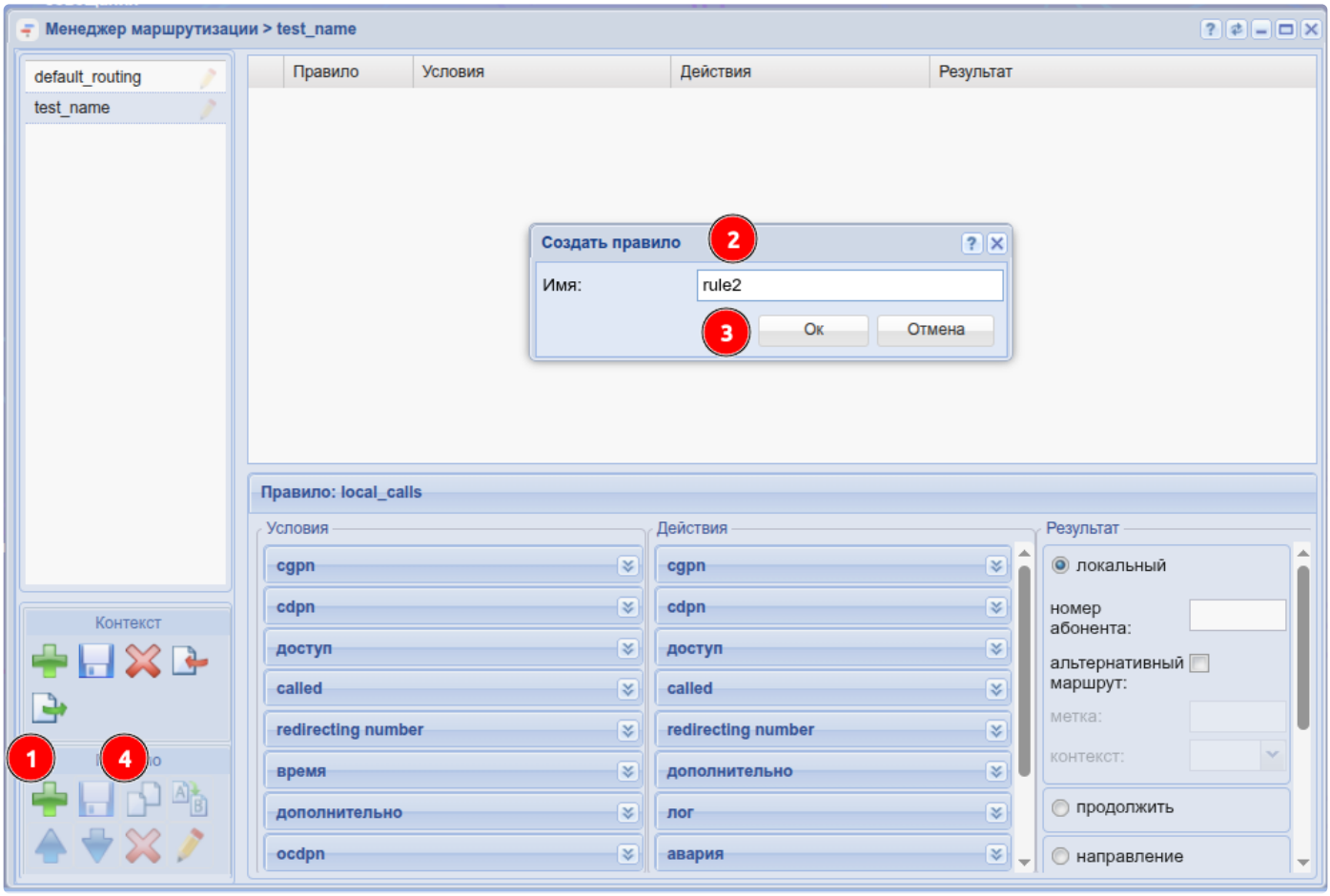


Рисунок 27 – Создание правила контекста маршрутизации

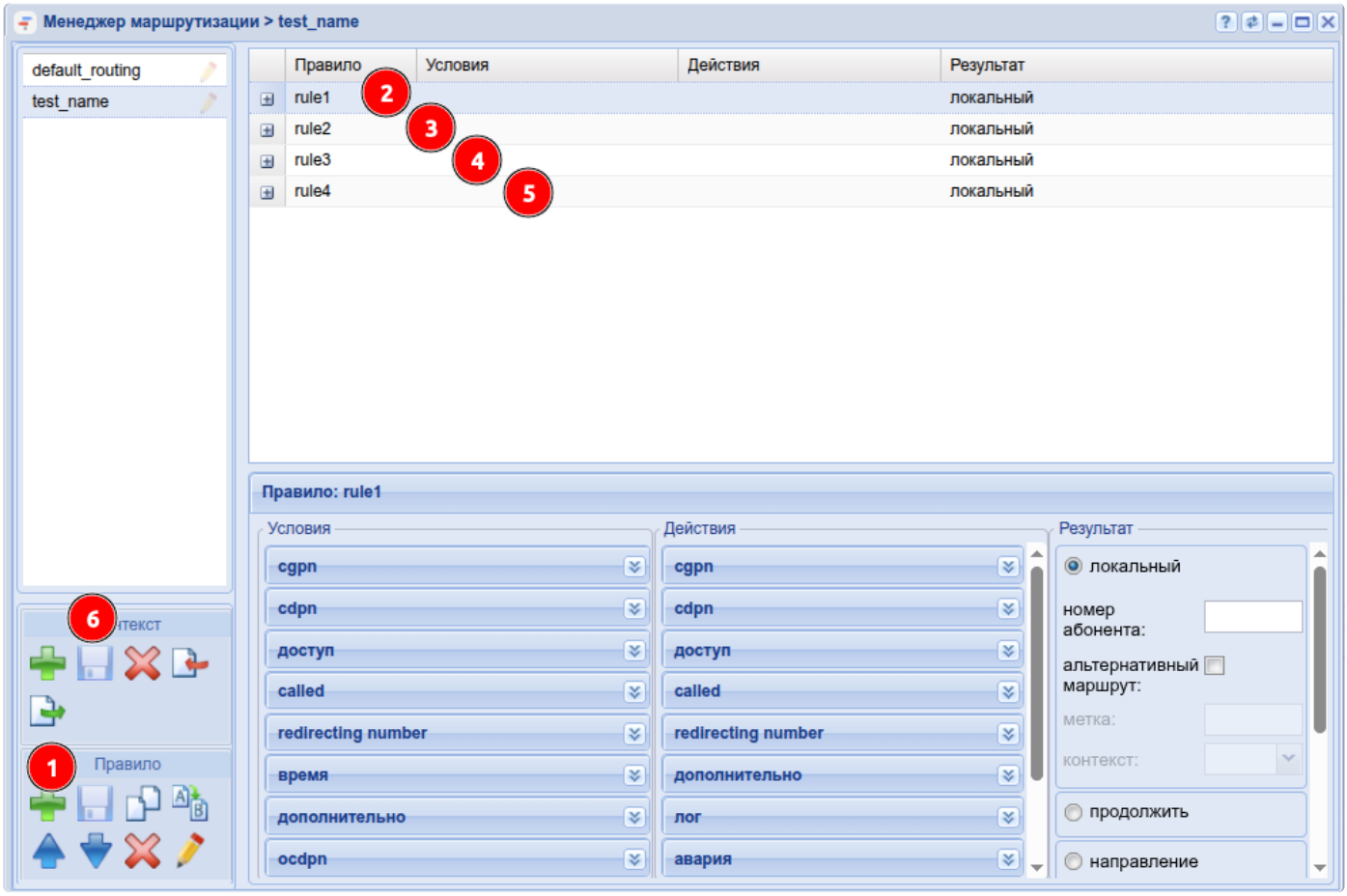


Рисунок 28 — Определение правил

На данный момент не определен транк, чтобы можно было сослаться на него в правиле, однако можно задать номера, по которым будет производится выборка.

Нажав на правило rule1, перейдем в нижнюю часть экрана, в которой располагаются области для редактирования контекста маршрутизации. Условно определим, что выборка для выхода на транк будет проводиться по признакам номера вызываемого абонента (CDPN), а номера в транке должны начинаться с цифры 4.

Функционально контекст маршрутизации делится на три части:

- Условия ("Condition") — раздел, определяющий выражения для выборки по предложенным признакам;
- Действия ("Action") — раздел, выполняющий преобразование признаков номеров к конкретному значению;
- Результат ("Result") — раздел, завершающий маршрутизацию и определяющий её результат.

rule1: Для выхода на транк нужно правильно отредактировать каждую часть:

- В разделе условий перейти на вкладку CDPN. Ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера с 42006 по 42007, нужно ввести условие: 420(6-7);
- В разделе действий перейти на вкладку CDPN. Ввести маску для изменения номера в поле "Номер". Например, чтобы добавить число 4 перед номером, нужно заполнить поле следующим выражением: 4{1,2,3,4,5};
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить транк в системе, поэтому к настройке данного правила вернемся чуть позже.

Таким же образом нужно настроить оставшиеся правила.

Для rule2:

- В разделе условий, во вкладке CDPN ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера со 42001 по 42005, ввести условие 420(1-5);
- В разделе результата определить результат как локальный (т.е. локальная маршрутизация);
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Для rule 3 допустим, что абоненты с номерами 108, 109 и 110 попадают в ivr-скрипт–информатор, прежде чем звонить далее.

- В разделе условий, во вкладке CGPN ввести номера телефонов, по которым будет выборка в поле "Номер", например 420(10,08-09).
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить в системе ivr-скрипт, поэтому к настройке данного правила вернемся чуть позже.

Для rule 4 мы определяем правило-исключение — это правило, которое работает в случае, если любые другие правила не срабатывают.

- По умолчанию данное правило создается в системе в контексте default_routing. В условиях на вызываемый номер прописывается символ %, а результатом является локальная маршрутизация, однако, если мы назначаем абоненту контекст, отличный от default_routing, в конце *рекомендуется* создавать это правило.

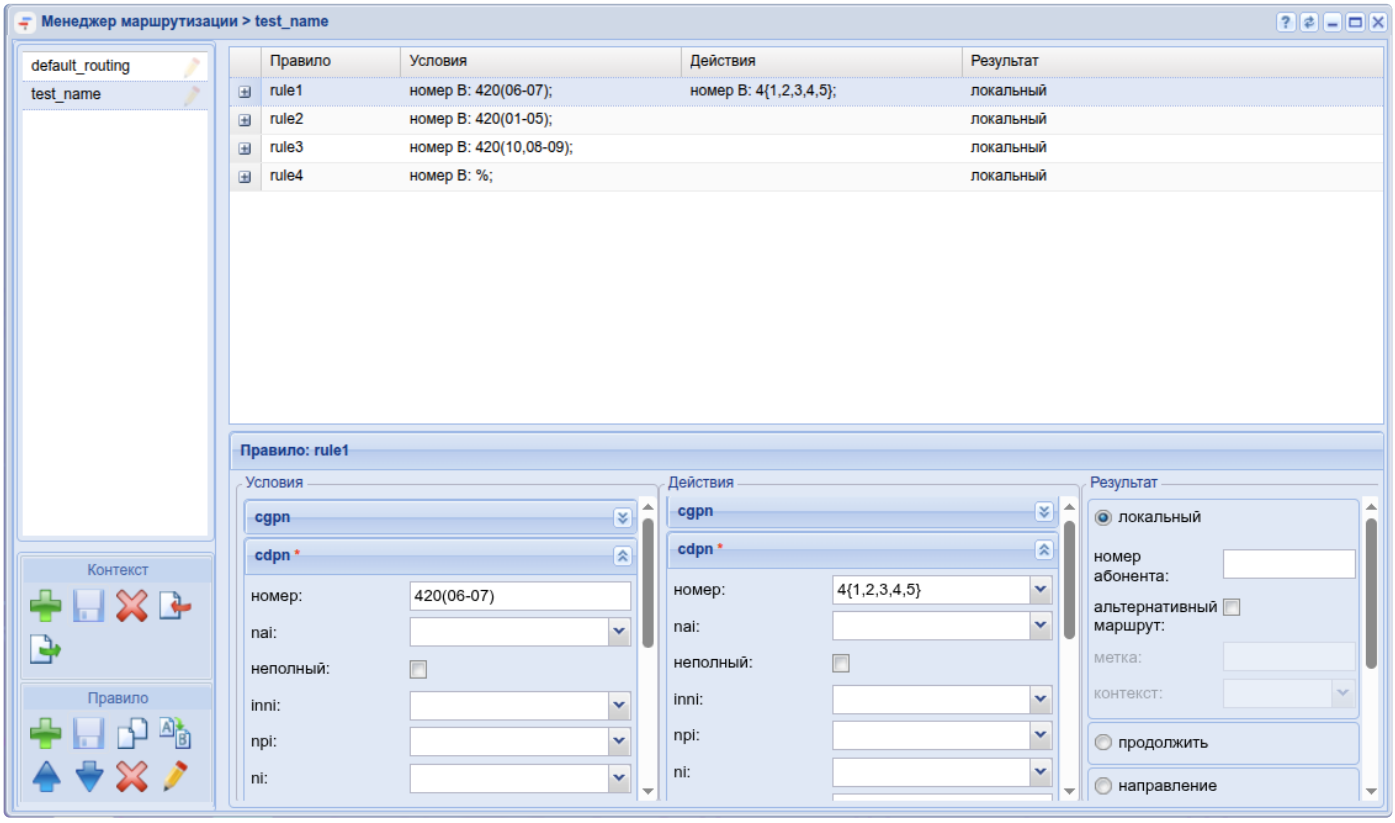


Рисунок 29 — Пример настроек контекста маршрутизации

10.6.9 Создание транка

Для создания и редактирования параметров транка в системе используется приложение "Менеджер транков" ("Trunk manager") .

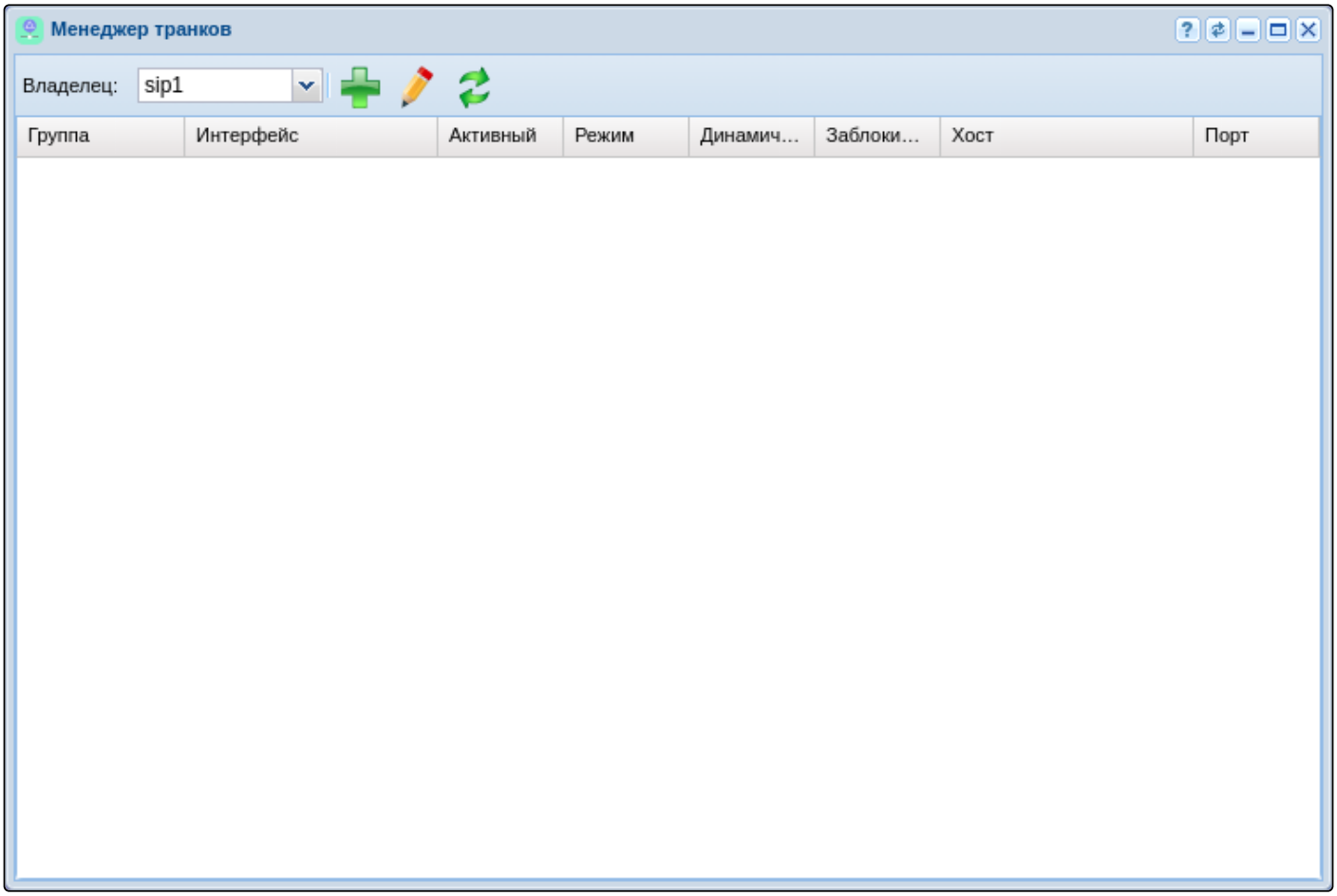


Рисунок 30 — Вид приложения "Менеджер транков"

Для определения транка в системе нужно нажать на кнопку *"Создать транк"* , в появившемся диалоговом окне определить параметры:

- *Имя* — назначить название транка, по которому его можно идентифицировать в системе;
- *Контекст* — применить ранее созданный контекст маршрутизации test_name;
- *Группа* — выбрать группу интерфейсов, созданную при определении абонентов "test.group";
- *Группа IP-адресов (IP-set)* — созданная на домене группа адресов "test_set";
- *Регистрация* — если используется транковая, то нужно включить данный параметр, в примере она не используется;
- *Хост [:порт]* — IP-адрес назначения транка — 10.0.3.100;
- *Порт для приёма* — транспортный порт, на котором будет слушаться трафик с транка, соответствует порту, назначенному на группу IP-адресов.

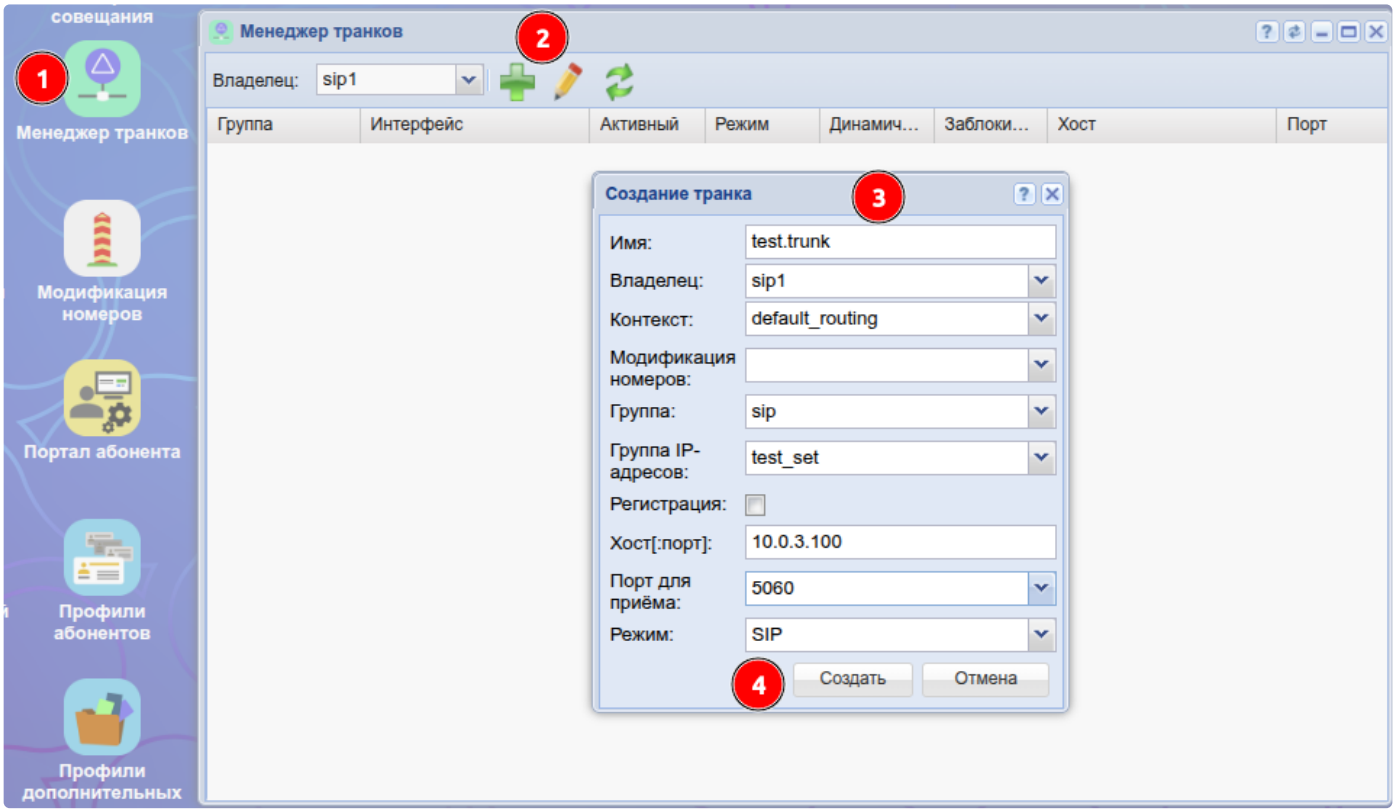


Рисунок 31 — Создание транка

10.6.10 Создание IVR-скрипта

Для создания IVR используется приложение "IVR-редактор" ("IVR editor").

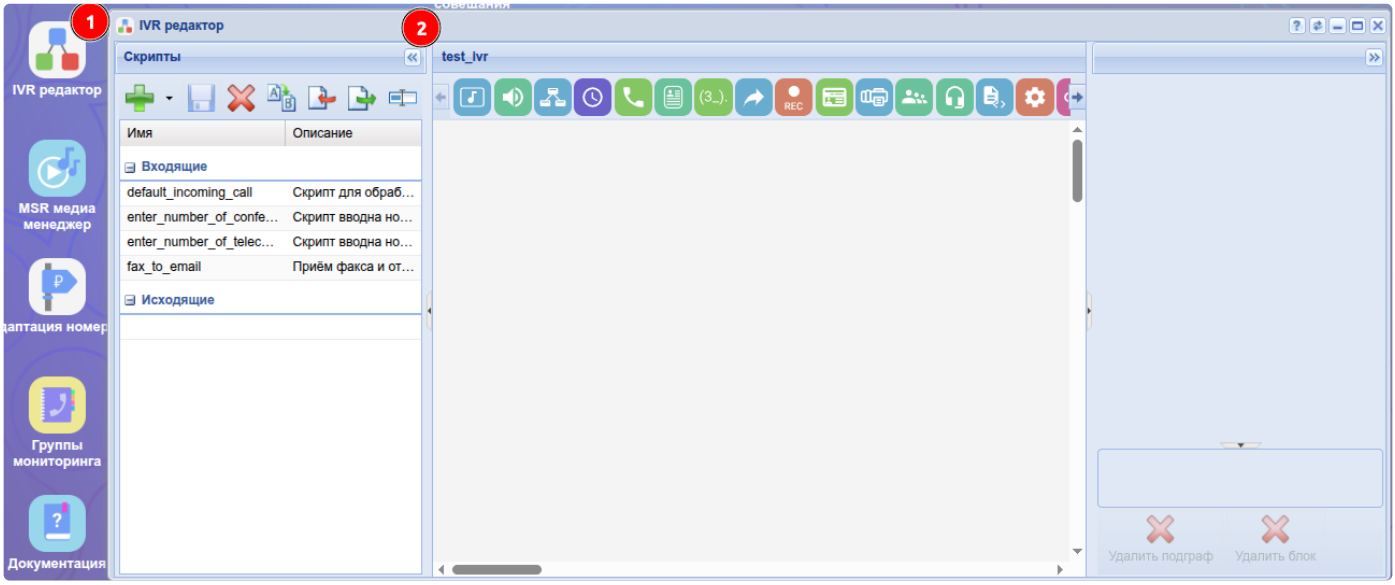


Рисунок 32 — Вид приложения "IVR-редактор"

Для создания скрипта нужно нажать на кнопку *"Добавить"* , выбрать тип скрипта (в данном случае будет скрипт на входящие вызовы), в диалоговом окне указать название скрипта, например "test_ivr".

После создания скрипта в основном окне редактора появится блок-схема.

В качестве примера составим скрипт, который при срабатывании проигрывает вызывающему абоненту заранее записанную фразу, а затем продолжает вызов.

Пример скрипта приведен на рисунках ниже. Здесь используются блоки Info и Dial.

- Блок Info — блок, который проигрывает сообщения до момента ответа пользователя. В качестве записанной фразы используется генератор тонов:

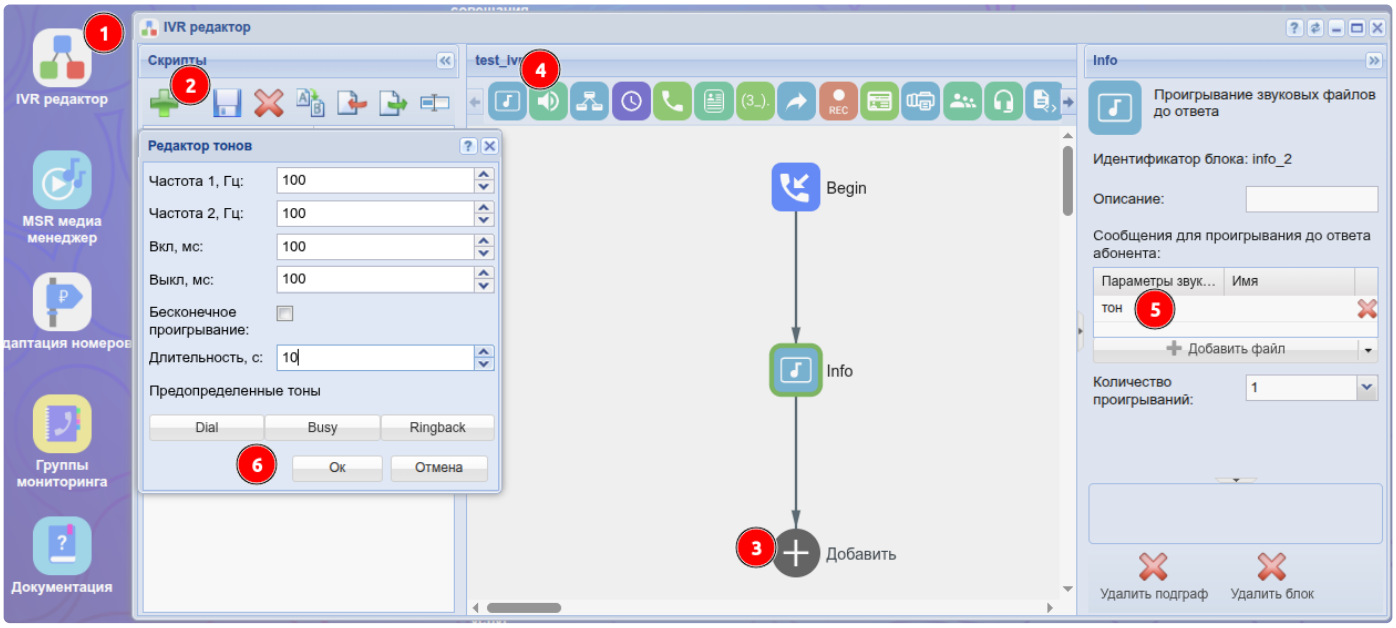


Рисунок 33 — Вид рабочей области IVR с настройками блока "Info"

- Блок dial — блок, который совершает вызов на заданный номер. Чтобы продолжить вызов, нужно применить предопределенную переменную CDPN:

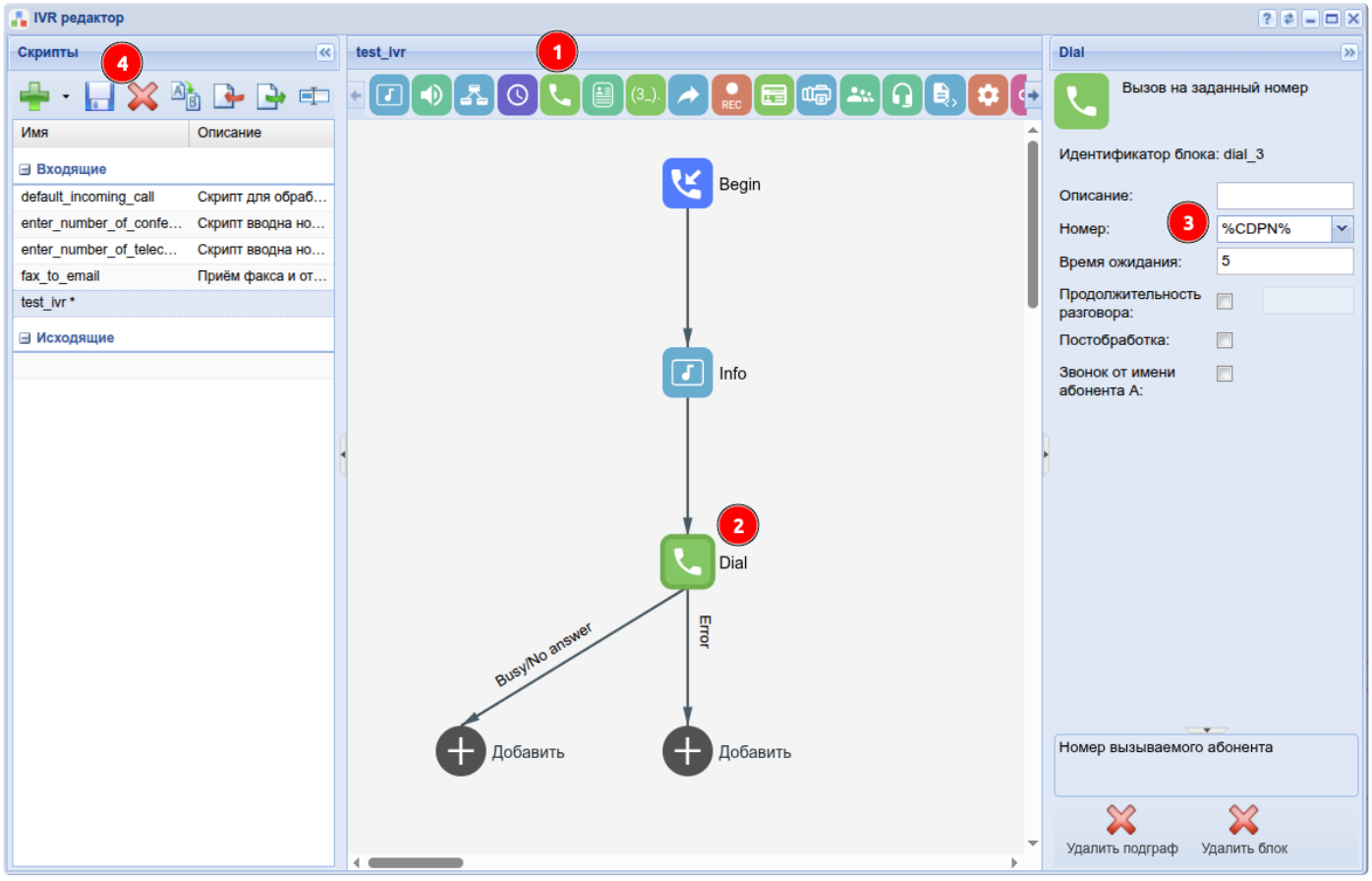


Рисунок 34 — Вид рабочей области IVR

10.6.11 Завершение настройки маршрутизации

Для завершения настройки маршрутизации нужно войти в приложение "Менеджер маршрутизации" ("Routing manager") и в правилах rule1 и rule3 настроить соответствующие результаты маршрутизации, для этого:

- rule1: в разделе "Результат" выберите подпункт "внешний". В таблице "Направления" в поле "Значение" добавьте созданный ранее транк, после чего сохраните правило .
- rule3: В разделе "Результат" выберите подпункт "ivr" и в поле скрипт добавьте созданный ранее ivr-скрипт, после чего сохраните правило и контекст .

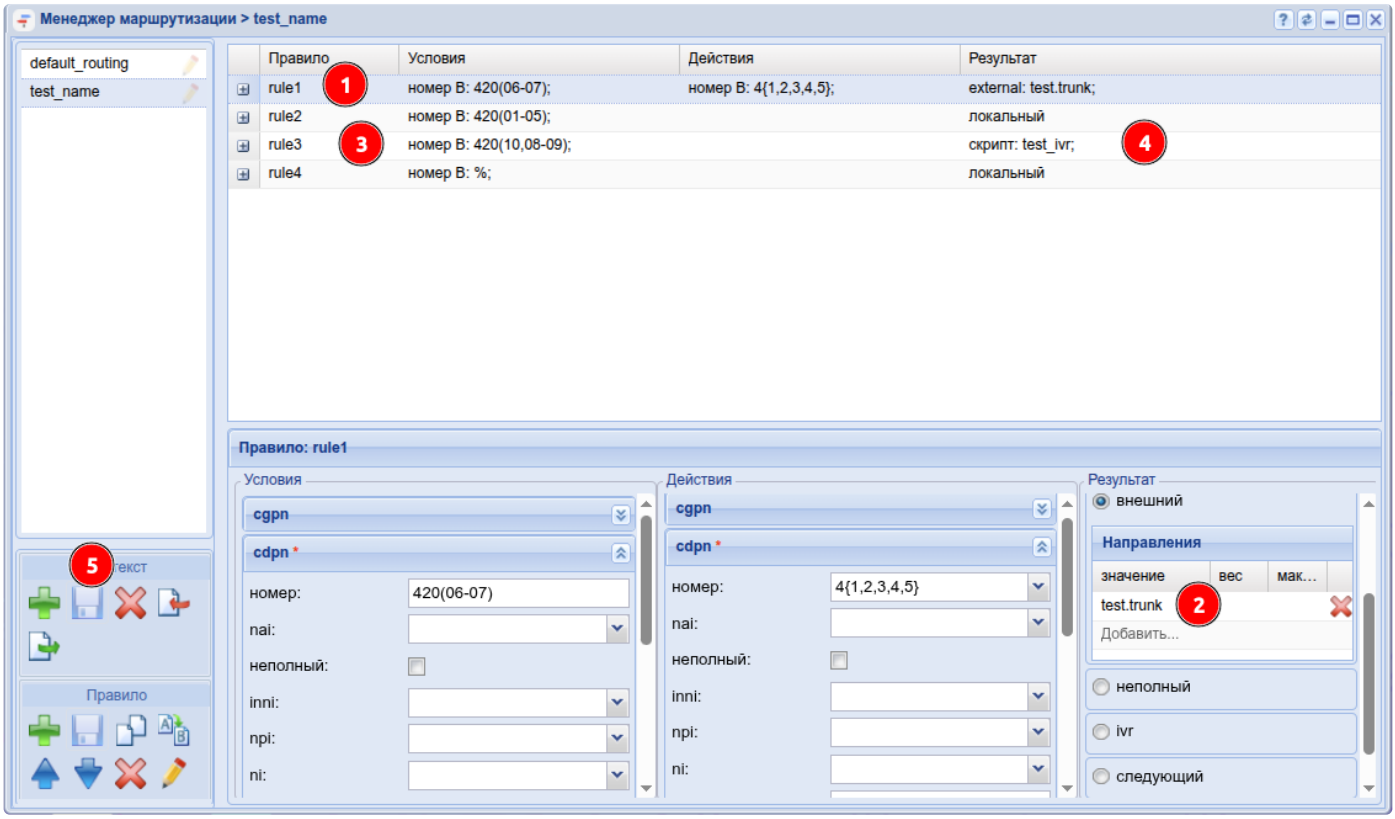


Рисунок 35 — Вид контекста маршрутизации

10.7 Первоначальная конфигурация системы через интерфейс командной строки (CLI)

пример первоначальной конфигурации через CoCon (CLI)

10.7.1 Добавление медиа-ресурсов в интерфейсе командной строки (CLI)

⚠ Для возможности использования медиаресурса и установления соединений , необходимо выполнить активизацию в команде создания медиаресурса - **/system/media/resource/declare** (параметр **"is active"=true**) или **/system/media/resource/set** (параметр **"active"=true**), если при создании активизация не была выполнена. Так же можно выполнить активизацию через веб интерфейс , приложение - сетевые окончания MSR (MSR-registrars).

Зайдите в CLI - CoCon:

```
ssh admin@localhost -p 8023
password: password
```

Медиаресурс — это описание параметров медиасервера, необходимых для работы с ним.

Медиаресурс — это описание параметров медиасервера, необходимых для работы с ним.

declare
Команда для объявления подключения медиасервера к ECSS-10.

Для работы в режиме предварительно настроенной системы можно заранее прописать медиаресурсы, информация о которых известна.

Для этого необходимо создать описание известных медиасерверов с указанием всех параметров, что позволить начать работу с медиасервером сразу после его регистрации.

Команда для создания конфигурации медиасервера:

Путь команды:

/system/media/resource/declare

Синтаксис:

declare <node> by_contact <contact> <zone> <site> <is active>

или

declare <node> by_iface <msr> <acc_id> <iface> <zone> <site> <is active>

Параметры:

<node> — имя ноды core;
<contact> — contact uri декларируемого медиаресурса;
<zone> — логическая зона обслуживания MSR;
<site> — идентификатор сети обслуживания MSR;
<is active> — состояние активности медиаресурса, значения true/false;
<msr> — имя MSR;
<acc_id> — номер идентификатор ресурса;
<iface> — название интерфейса.

Установить связь MSR- и Core-подсистем. Для этого выполнить команды:

```
/system/media/resource/declare core1@ecss1 contact net.20@msr.ecss1  default local true
```

```
/system/media/resource/declare core1@ecss2 contact net.20@msr.ecss2  default local true
```

```
/system/media/resource/declare core1@ecss1 contact net.20@msr.ecss2  default local true
```

```
/system/media/resource/declare core1@ecss2 contact net.20@msr.ecss1  default local true
```

результат

```
/system/media/resource/info core1@ecss1 net.20@msr.ecss1
  Media resource specific for node core1@ecss1:
┌─┬──────────────────┬──────────────────────────┐
│F│    Parameter     │          Value           │
├─┼──────────────────┼──────────────────────────┤
│r│ Contact          │ net.20@msr.ecss1         │
│r│ Status           │ registered               │
│r│ Cc-status        │ connected                │
│r│ Cc-status uptime │ 00:39:35                 │
│r│ Cc-id            │ db21af                   │
│r│ Address          │ 10.0.20.51:5700          │
│r│ Register time    │ 2025/12/02 16:43:07      │
│r│ Expired          │ 84                       │
│r│ Iface            │ net.20(10.0.20.51)       │
│ │ Active           │ true                     │
│ │ Zone             │ default                  │
│ │ Site             │ local                    │
│r│ MSR version      │ 3.18.1.0.10              │
│r│ MSR load         │ 0                        │
│r│ MSR perf coef    │ 1.0                      │
└─┴──────────────────┴──────────────────────────┘

Legend:
Column F - means Flag, values r - flag for readonly parameter
```

результат / список

```
/system/media/resource/list all
  All media resource selected list specific:
┌──────────────┬────────────┬──────────────┬──────────┬──────┬────────┬──────────┬──────────
──┬──────────┬──────────┐
│     Node     │    MSR     │     MSR      │    MSR   │ MSR  │ Cc-id  │ Cc-status │ Cc-uptime
│   Cc-address │ Iface  │   Iface    │ Active │  Zone   │ Site  │     Contact      │
   Status   │ Expired │
│              │            │   version    │ perf coef │ load │        │           │
│                  │  name  │    addr    │          │          │       │                  │
│          │
├──────────────┼────────────┼──────────────┼──────────┼──────┼────────┼──────────┼──────────
──┼──────────┼──────────┤
│ core1@ecss1 │ msr.ecss1 │ 3.18.1.0.10 │       1.0 │    0 │ db21af │ connected │ 00:40:22
│ 10.0.20.51:5700 │ net.20 │ 10.0.20.51 │ true   │ default │ local │ net.20@msr.ecss1 │
 registered │     156 │
│              │ msr.ecss2 │ 3.18.1.0.10 │       1.0 │    0 │ 263898 │ connected │ 00:40:22
│ 10.0.20.52:5700 │ net.20 │ 10.0.20.52 │ true   │ default │ local │ net.20@msr.ecss2 │
 registered │     156 │
```

```
| core1@ecss2 | msr.ecss1 | 3.18.1.0.10 |       1.0 |    0 | db21af | connected | 00:41:11
| 10.0.20.51:5700 | net.20 | 10.0.20.51 | true   | default | local | net.20@msr.ecss1 |
 registered |     106 |
|             | msr.ecss2 | 3.18.1.0.10 |       1.0 |    0 | 263898 | connected | 00:44:00
| 10.0.20.52:5700 | net.20 | 10.0.20.52 | true   | default | local | net.20@msr.ecss2 |
 registered |     177 |
└─────────────┴──────────┴─────────────┴──────────┴──────┴────────┴───────────┴──────────
┴──────────────────┴────────┴─────────────┴────────┴─────────┴──────┴──────────────────┴──────
────────┴──────────┘
```

10.7.2 Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI)

/system/media/registrar/ - команды управления регистратором для MSR медиа менеджера

/system/media/registrar/ - команды управления регистратором для MSR медиа менеджера

set

Команда для изменения общих и индивидуальных параметров.

Путь команды:

/system/media/registrar/set

Синтаксис настройки общих параметров:

set common <FIELD> <VALUE>

Параметры:

<FIELD> - имя настраиваемого параметра;

<VALUE> - значение настраиваемого параметра

Где

- common - тип выставляемого параметра;
- <common parameter> - общий параметр медиарегистратора для всех нод: registration-timeout;
- private - тип выставляемого параметра;
- <node name> - имя ноды на которой находится медиарегистратор;
- <private parameter> - параметр медиарегистратора, который выставляется на <node name>: is-active, listen-interface & listen-port.

Список параметров доступных для настройки:

- is-active - позволяет включить или выключить регистратор (false - деактивирует регистратор, регистратор перестает принимать SIP трафик);
- listen-interface - имя интерфейса, который использует регистратор;
- listen-port - порт, на котором регистратор принимает запросы на регистрацию (по умолчанию 5000);
- registration-timeout - таймер перерегистрации, диапазон значений от 60 до 86400.

В общих параметрах доступна настройка времени регистрации, по умолчанию установлено 60 секунд.

Пример:
Установить время регистрации 60 секунд:

```
/system/media/registrar/set common registration-timeout 60
Common parameter registration-timeout set to 60.
```

Синтаксис настройки индивидуальных параметров:

system/media/registrar/set private <CORE> <FIELD> <VALUE>

Параметры:

<CORE> - нода ядра, для которой настраивается Registrar-сервер;

<FIELD> - имя настраиваемого параметра;

<VALUE> - значение настраиваемого параметра.

Настраиваемые параметры:

- is-active - активен ли Registrar-сервер, принимает значения true | false;
- listen-ip - ip-адрес, на котором Registrar-сервер будет принимать соединения (по умолчанию 0.0.0.0);
- listen-port - сетевой порт, на котором Registrar-сервер будет принимать соединения (по умолчанию 5000).

Установить ipадрес для приема соединений от медиасервера:

выполняется в CoCon

```
/system/media/registrar/set private core1@ecss1 listen-ip 10.0.20.41
```

```
/system/media/registrar/set private core1@ecss1 listen-ip 10.0.20.41
Private parameter listen-ip set to 10.0.20.41 at node core1@ecss1.
```

выполняется в CoCon

```
/system/media/registrar/set private core1@ecss2 listen-ip 10.0.20.42
```

```
/system/media/registrar/set private core1@ecss2 listen-ip 10.0.20.42
Private parameter listen-ip set to 10.0.20.42 at node core1@ecss2.
```

Результат:

```
/system/media/registrar/info
 Common parameters:
┌──────────────────────────────┬──────┐
```

```
|         Parameter          | Value |
|----------------------------|-------|
| registration-timeout, sec  |   180 |
```

Private parameters:

```
|    Node      | Status  | Is-active | Listen-ip   | Listen-port |
|--------------|---------|-----------|-------------|-------------|
| core1@ecss1  | active  | true      | 10.0.20.41  |        5000 |
| core1@ecss2  | active  | true      | 10.0.20.42  |        5000 |
```

После того как все сервисы будут запущены, ноды какое-то время будут устанавливать связь. Как только все ноды загрузятся , при выполнении запроса **system-status**, последует вывод следующей информации:

```
/system-status
Checking...
|  |     Node       |          Release          |         Erlang nodes          |    Mnesia
nodes     |Uptime |
|--|----------------|---------------------------|-------------------------------|-----------
---------|-------|
|  |core1@ecss1     |ecss-core-3.18.1.0.1656    |core1@ecss1,core1@ecss2        |not running
|43m 43s|
|  |core1@ecss2     |ecss-core-3.18.1.0.1656    |core1@ecss1,core1@ecss2        |not running
|47m 5s |
|  |ds1@ecss1       |ecss-ds-3.18.1.0.1656      |ds1@ecss1,ds1@ecss2
|ds1@ecss1,ds1@ecss2  |43m 43s|
|  |ds1@ecss2       |ecss-ds-3.18.1.0.1656      |ds1@ecss1,ds1@ecss2
|ds1@ecss1,ds1@ecss2  |47m 4s |
|  |md1@ecss1       |ecss-mediator-3.18.1.0.1656|md1@ecss1,md1@ecss2
|md1@ecss1,md1@ecss2  |43m 43s|
|  |md1@ecss2       |ecss-mediator-3.18.1.0.1656|md1@ecss1,md1@ecss2
|md1@ecss1,md1@ecss2  |47m 5s |
|  |mycelium1@ecss1|ecss-mycelium-3.18.1.0.1656|mycelium1@ecss1,mycelium1@ecss2|not running
|43m 43s|
|  |mycelium1@ecss2|ecss-mycelium-3.18.1.0.1656|mycelium1@ecss1,mycelium1@ecss2|not running
|47m 4s |
|  |sip1@ecss1      |ecss-pa-sip-3.18.1.0.1656  |sip1@ecss1,sip1@ecss2
|sip1@ecss1,sip1@ecss2|43m 43s|
|  |sip1@ecss2      |ecss-pa-sip-3.18.1.0.1656  |sip1@ecss1,sip1@ecss2
|sip1@ecss1,sip1@ecss2|47m 5s |
|  |                |                           |                               |
---------|-------|

All services are started.

  Active media resource selected list specific:
|    Node      |    MSR     |     MSR      | Cc-status | Cc-uptime |
|              |            |   version    |           |           |
|--------------|------------|--------------|-----------|-----------|
| core1@ecss1 | msr.ecss1 | 3.18.1.0.10 | connected | 00:42:49  |
```

```
|              | msr.ecss2 | 3.18.1.0.10 | connected | 00:42:48  |
| core1@ecss2 | msr.ecss1 | 3.18.1.0.10 | connected | 00:43:38  |
|              | msr.ecss2 | 3.18.1.0.10 | connected | 00:46:27  |
└──────────────┴─────────────┴──────────────┴─────────────┴─────────────┘

Alarms list on md1:
┌──────────────┬──────────────┬──────────────┬──────────────────────┬──────────────────────┐
└──────────────┴──────────────┴──────────────┴──────────────────────┴──────────────────────┘
|     Date     |   Severity   |   Location   |       Cause          |       Class          |
     Instance      |                          Message                          |
├──────────────┼──────────────┼──────────────┼──────────────────────┼──────────────────────┤
└──────────────┴──────────────┴──────────────┴──────────────────────┴──────────────────────┘

Selected: 0
Total: 0
```

В примере выше видно, что ноды вошли в кластер и *MSR* зарегистрировались на ноде *ecss-core*.

Создание и настройка группы SIP-IP-адресов (IP-set)

IP-set — совокупность одного или нескольких IP-адресов из одной подсети и списка слушающих UDP/TCP-портов, которые будут открыты SIP-адаптером на данных адресах.

Произведите настройку адаптера SIP согласно ТЗ (команды выполнить в CoCon):

выполняется в CoCon

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss1 ip = 10.0.20.31
```

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node =
sip1@ecss1 ip = 10.0.20.31
Property "ip_set" successfully changed from:

to
test_set: no ports set
test_set: sip1@ecss1 10.0.20.31
test_set: dscp 0.
```

выполняется в CoCon

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss2 ip = 10.0.20.32
```

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node =
sip1@ecss2 ip = 10.0.20.32
Property "ip_set" successfully changed from:
test_set: no ports set
test_set: sip1@ecss1 10.0.20.31
test_set: dscp 0
to
test_set: no ports set
test_set: sip1@ecss1 10.0.20.31
test_set: sip1@ecss2 10.0.20.32
test_set: dscp 0.
```

выполняется в CoCon

```
/cluster/adapter/sip1/sip/network/set ip_set test_set listen-ports list = [5060]
```

```
/cluster/adapter/sip1/sip/network/set ip_set test_set listen-ports list = [5060]
Property "ip_set" successfully changed from:
```

```
test_set: no ports set
test_set: sip1@ecss1 10.0.20.31
test_set: sip1@ecss2 10.0.20.32
test_set: dscp 0
   to
test_set: 5060
test_set: sip1@ecss1 10.0.20.31
test_set: sip1@ecss2 10.0.20.32
test_set: dscp 0
```

10.7.3 Создание Домена и подключение IP-set

Создайте Домен (виртуальную АТС) и присвойте ему созданную выше группу (IP-set) настроек SIP-адаптера (команды выполняются в CoCon):

выполняется в CoCon

```
/domain/declare test_domain --add-domain-admin-privileges --add-domain-user-privileges
```

```
/domain/declare test_domain --add-domain-admin-privileges --add-domain-user-
privileges
New domain test_domain is declared
```

выполняется в CoCon

```
/domain/test_domain/sip/network/set ip_set [test_set]
```

```
/domain/test_domain/sip/network/set ip_set [test_set]

Property "ip_set" successfully changed from:

[]
    to
["test_set"].
```

10.7.4 Добавление услуг

Разрешить оператору Домена (виртуальной АТС) "test_domain" управлять всеми услугами :

выполняется в CoCon

```
/cluster/storage/ds1/ss/access-list add test_domain *
```

```
/cluster/storage/ds1/ss/access-list add test_domain *
Supplementary services successfully added in the domain test_domain.
```

Для возможности применения пакета услуг у абонентов сразу после создания, необходимо настроить политику применения лицензионных пакетов услуг (детальную информацию вы можете посмотреть здесь)

Создать политику применения лицензии с именем -"new_sub"

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/declare new_sub "Задаются при создании абонента"
```

Задать имя пакета услуг дополнительных сервисов, для этого использовать следующие команды :

выполняется в CoCon user/pw admin/password

```
/cluster/storage/ds1/licence/show-licence 1
```

нас интересует следующая информация :

```
/cluster/storage/ds1/licence/show-licence 1
SSW ID                                                  < Имя SSW ID >
Description                                             Лицензия для тестирования.

. . .

Supplementary Services licence package(s)
 name                                                   'ECSS-FULL+'
 limit                                                  600000
 description                                            "Весь пакет услуг"
 SS list                                                [1,2,3,4,5]

. . .
```

имя пакета услуг в примере "ECSS-FULL+".
Добавить в созданную политику пакет лицензионных услуг с именем "ECSS-FULL+", именно так он описан в лицензии:

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/package-add new_sub ECSS-FULL+
```

Пакет добавлен, но в настоящий момент политика не активна (не может быть применена к абоненту). Для активации политики new_sub выполнить следующую команду:

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/activate new_sub
```

Теперь при создании абонентов можем использовать дополнительные услуги.

11 Установка ECSS10 3.18 на Astra Linux, в открытом контуре, на одиночный сервер

11.1 Первоначальная установка системы на одном сервере

11.1.1 Исходные данные

Конфигурация системы с одним сервером

Исходные данные

В данном разделе использованы специфичные для данного продукта термины, ознакомиться с ними можно здесь.
Техническое задание:
Требуется интеграция программного коммутатора 5 класса ECSS-10 (SSW) на 2-х физических серверах с поддержкой SIP со следующими параметрами на нагрузку группы серверов:

- Максимальное количество абонентов 15000 (MUL — Max user limit);
- Максимальное количество одновременных соединений 2000 (MCL — Max call limit);
- Требуется полный резерв системы ecss (все node резервируется на втором сервере);
- Количество сетевых интерфейсов ethernet — 3/5.

По техническому заданию требуется определить аппаратную платформу.

Полоса пропускания СПД

Требуемая полоса пропускания СПД **не менее** 1000Mb/s

Пример составления аппаратных требований:

Пример составления аппаратных требований:

Пример разнесения составляющих по адресному пространству

Имя сервера (хоста)	Роль	Интерфейс	Адрес/Маска	Порт
Статические адреса программного коммутатора				
ecss1	Адрес сервера	eth0	192.168.56.40/24	2000
ecss1	Адрес протокольного адаптера	int.20 (voip)	10.0.20.40/24	5000
ecss1	Адрес шлюза	eth0	192.168.56.1	-
ecss1	Адреса DNS-серверов	eth0	8.8.8.8	-
Виртуальные адреса программного коммутатора				

ecss1	Адрес ядра (*ecss-core*)	lo	10.0.20.40/24	5000
ecss1	Адрес медиа-сервера (*ecss-media-server (MSR)*)	lo	10.0.20.40/24	5000

Заполнить данные параметры для SSW из проектного технического задания				
Имя сервера (хоста)	**Роль**	**Интерфейс**	**Адрес/Маска**	**Порт**
Статические адреса программного коммутатора				
ecss1	Адрес сервера			2000
ecss1	Адрес протокольного адаптера			5000
ecss1	Адрес шлюза			-
ecss1	Адреса DNS-серверов			-
Виртуальные адреса программного коммутатора				
ecss1	Адрес ядра (*ecss-core*)	lo		5000
ecss1	Адрес медиа-сервера (*ecss-media-server (MSR)*)	lo		5000

Включение в сеть

Топологию включения сервера в сеть для обеспечения резервирования рекомендуется делать с использованием 2-х коммутаторов. Пример на рисунке ниже.

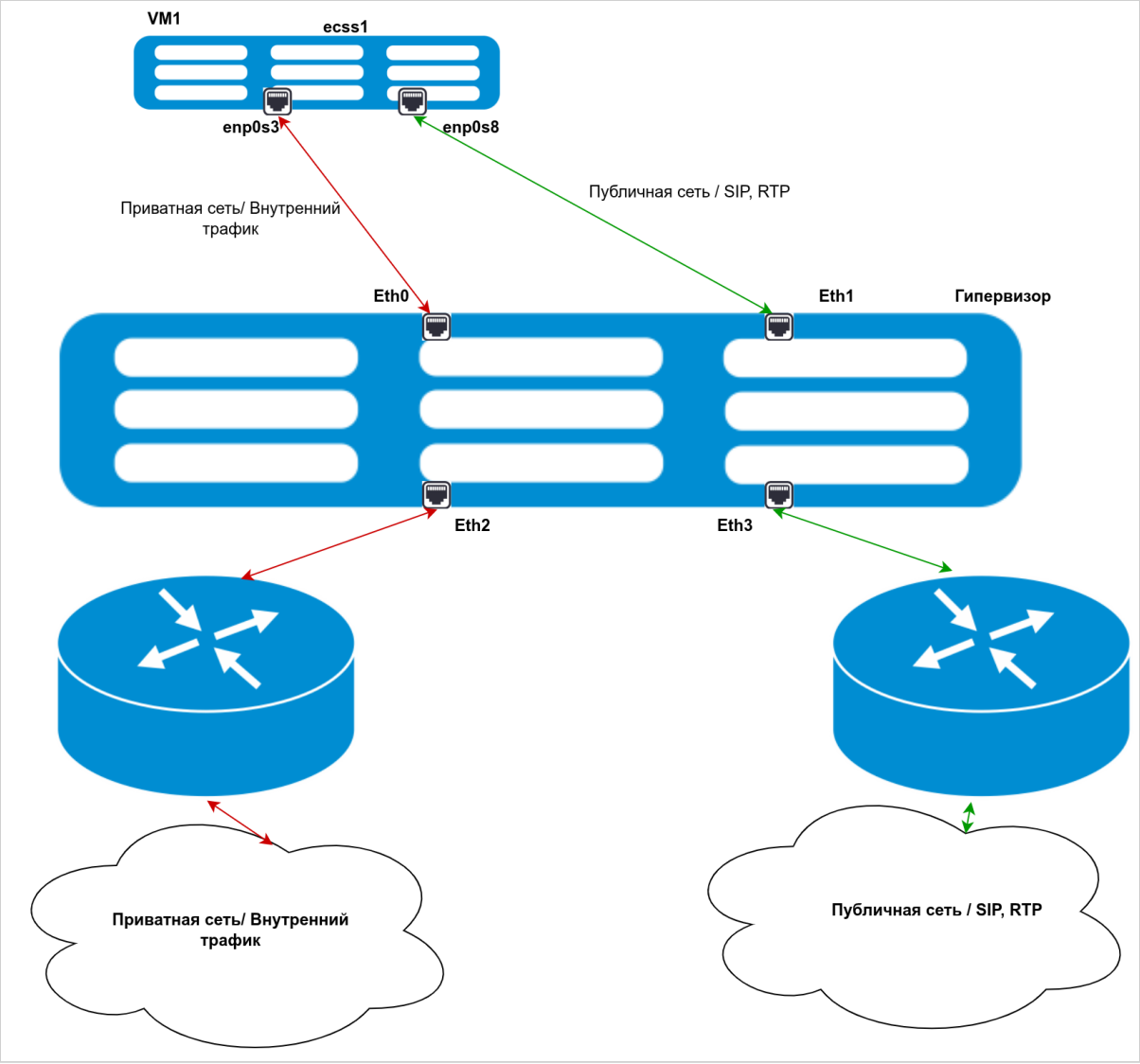


Рисунок 1 — Схема организации связи

11.1.2 Последовательность установки SSW ECSS10 без резервирования с одним сервером

Установка SSW ECSS10 состоит из 2-х основных частей :

- Подготовительной - Установка ОС Astra Linux 1.8, обновлений пакетов ОС Astra Linux 1.8 , "оптимизация" ОС, установка дополнительных пакетов программ, подготовка сетевых интерфейсов, /etc/hosts, ssh-keygen
- Установка приложений Eltex ECSS10

Подготовительный этап

В этом разделе приведено описание инсталляции операционной системы, а также необходимых и дополнительных пакетов. Система ECSS-10 версии 3.18 работает под управлением ОС Astra Linux 1.8.

Предварительные требования

- Установочный загрузочный носитель с дистрибутивом операционной системы ;

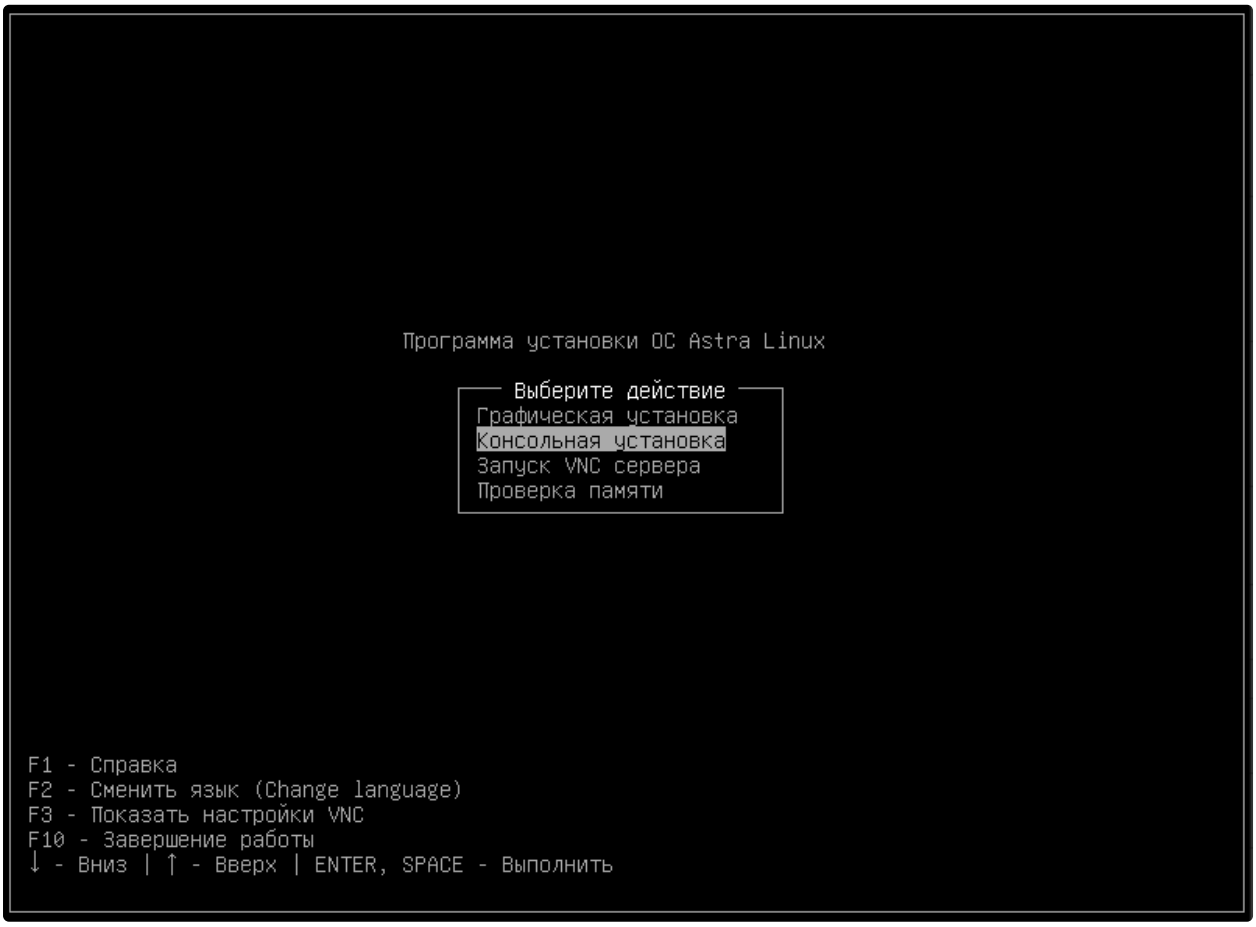
- Подготовленный сервер с обновленным BIOS, ILO (если есть), подключенная сеть для доступа в сеть Интернет;
- Выставленный первый приоритет загрузки с установочного носителя — USB Flash или CD/DVD в BIOS;
- Достаточный объем дискового пространства и памяти в соответствии с проектом.

11.1.3 Установка ОС

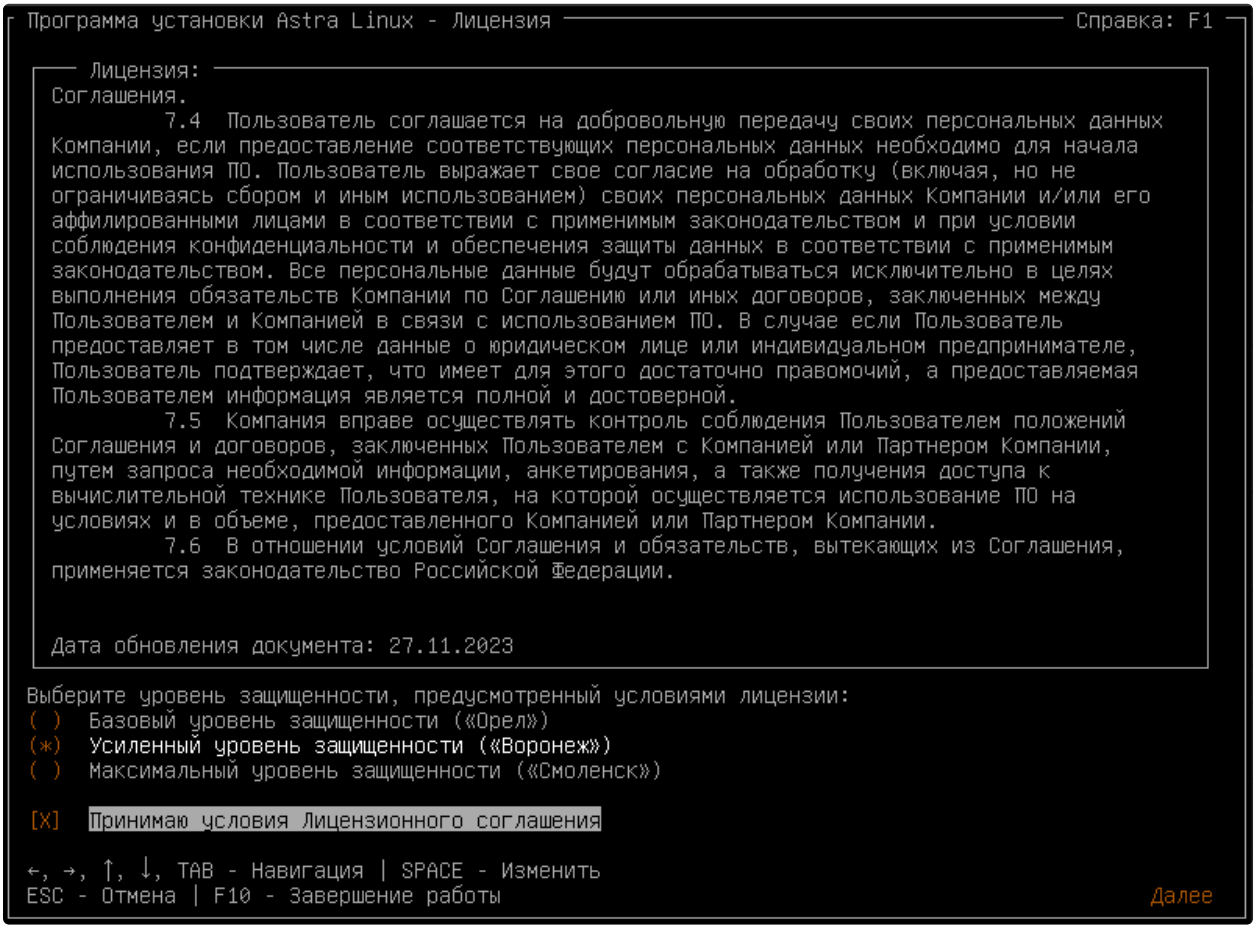
Для установки ОС необходимо выполнить следующее:

- После загрузки с установочного носителя выбрать "Графическая установка" или "Установка".
- Согласится с лицензионным соглашением.
- Выбрать раскладку клавиатуры.
- Выбрать сетевой интерфейс для подключения к сети Интернет.
- Установить имя компьютера (рекомендованы ecss1)
- Настроить пользователя- администратора.
- Создать партиции на диске в соответствии с таблицей 1 (разметка диска - Вручную).
- В процессе установке , **не создавать раздел swap**. При напоминании о пропуске раздела , **подтвердить** , что установка выполняется без создания swap.
- Установить часовой пояс.
- Установить OpenSSH server (активизировать пункт : "Средства удаленного подключения SSH").
- Выбрать уровень защищенности, который используется в вашем проекте (какая лицензия)
- Дополнительные настройки ОС - по умолчанию
- Перезагрузить успешно установленную ОС
  Установка ОС Astra Linux 1.8

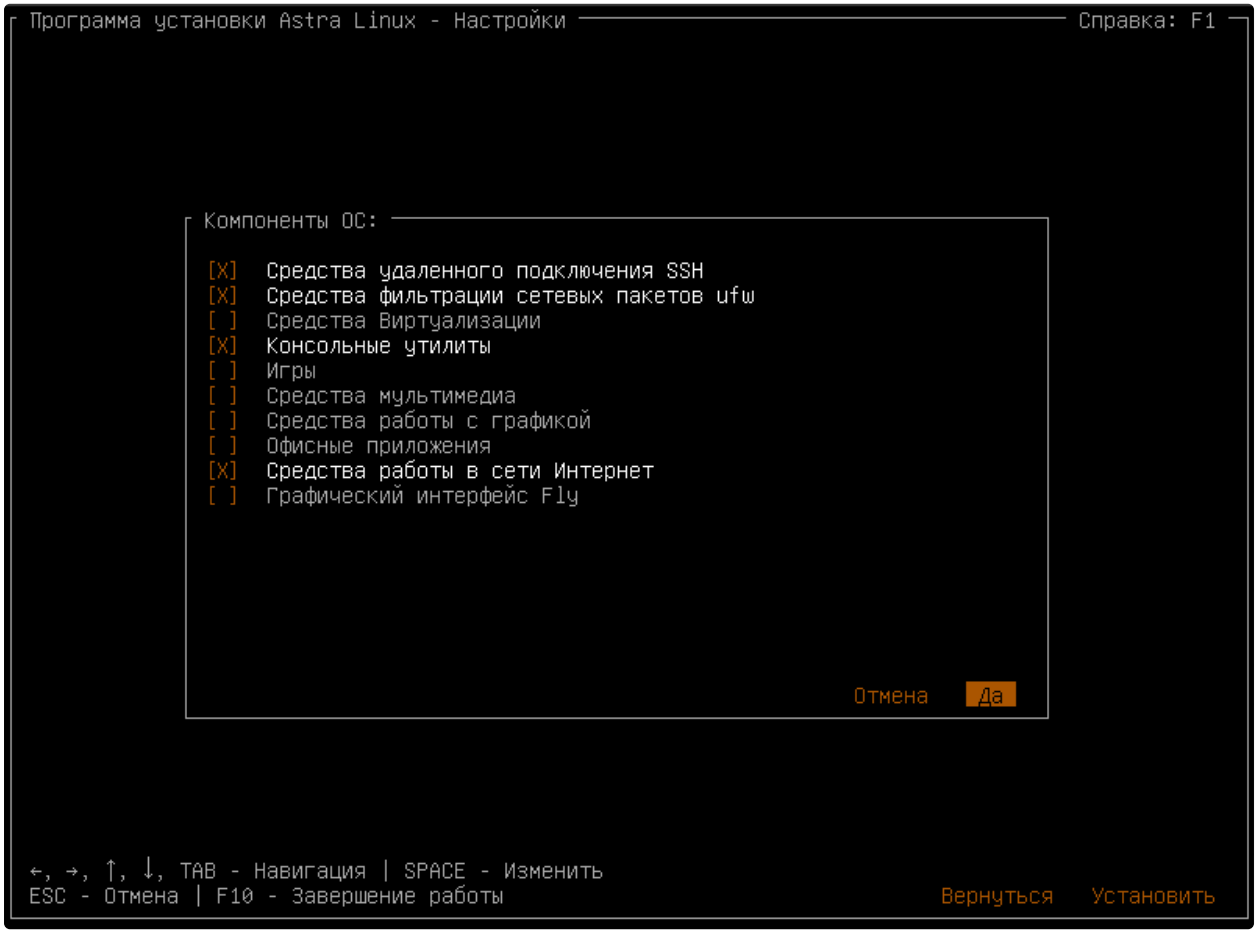
> **Выбрать режим установки**



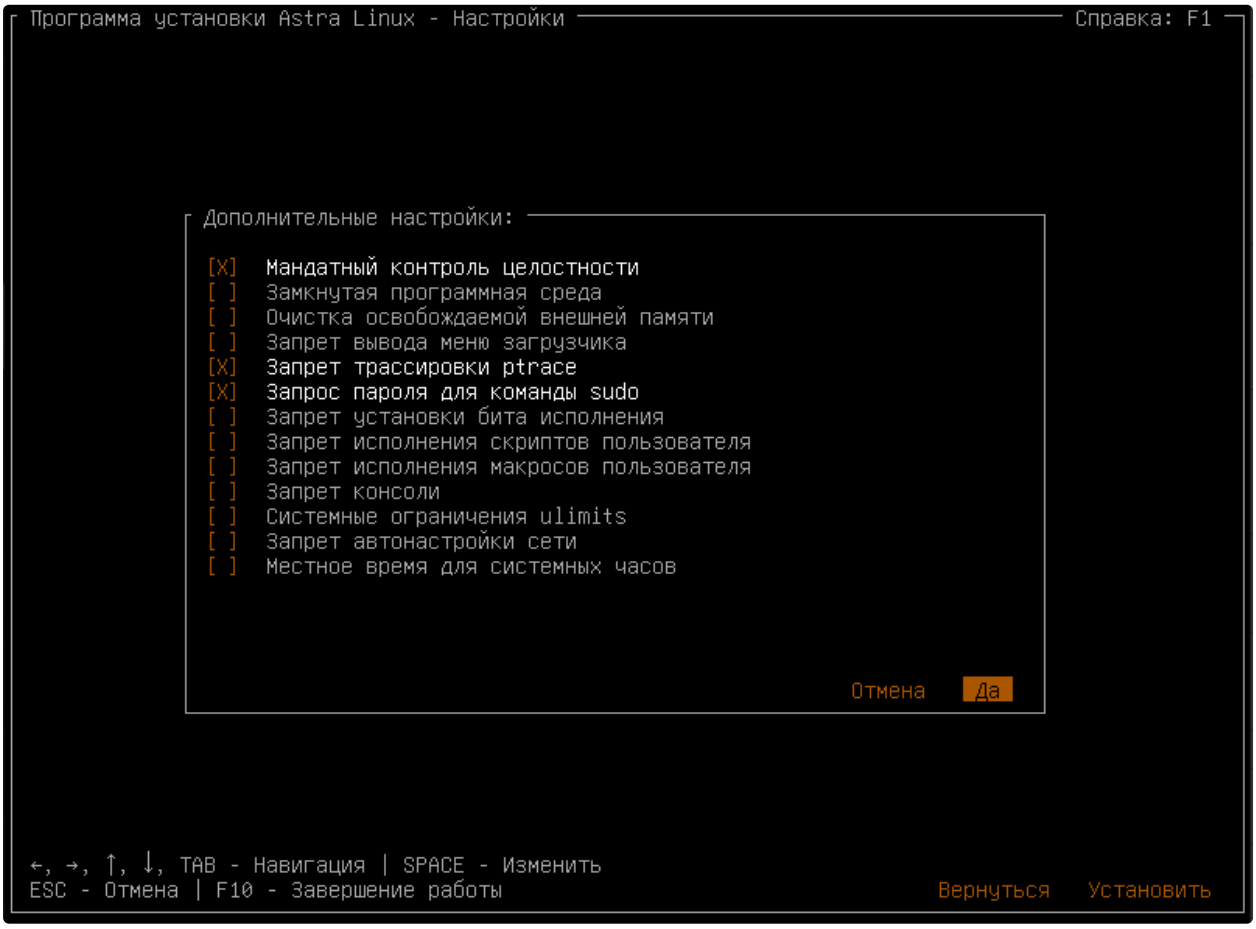
Согласится с Лицензионным Соглашением


Программа установки Astra Linux - Лицензия
Справка: F1
Лицензия:
Соглашения.
7.4 Пользователь соглашается на добровольную передачу своих персональных данных Компании, если предоставление соответствующих персональных данных необходимо для начала использования ПО. Пользователь выражает свое согласие на обработку (включая, но не ограничиваясь сбором и иным использованием) своих персональных данных Компании и/или его аффилированными лицами в соответствии с применимым законодательством и при условии соблюдения конфиденциальности и обеспечения защиты данных в соответствии с применимым законодательством. Все персональные данные будут обрабатываться исключительно в целях выполнения обязательств Компании по Соглашению или иных договоров, заключенных между Пользователем и Компанией в связи с использованием ПО. В случае если Пользователь предоставляет в том числе данные о юридическом лице или индивидуальном предпринимателе, Пользователь подтверждает, что имеет для этого достаточно правомочий, а предоставляемая Пользователем информация является полной и достоверной.
7.5 Компания вправе осуществлять контроль соблюдения Пользователем положений Соглашения и договоров, заключенных Пользователем с Компанией или Партнером Компании, путем запроса необходимой информации, анкетирования, а также получения доступа к вычислительной технике Пользователя, на которой осуществляется использование ПО на условиях и в объеме, предоставленного Компанией или Партнером Компании.
7.6 В отношении условий Соглашения и обязательств, вытекающих из Соглашения, применяется законодательство Российской Федерации.
Дата обновления документа: 27.11.2023
Выберите уровень защищенности, предусмотренный условиями лицензии:
( ) Базовый уровень защищенности («Орел»)
(*) Усиленный уровень защищенности («Воронеж»)
( ) Максимальный уровень защищенности («Смоленск»)
[X] Принимаю условия Лицензионного соглашения
←, →, ↑, ↓, TAB - Навигация | SPACE - Изменить
ESC - Отмена | F10 - Завершение работы
Далее

Настройки



Настойки (продолжение1)



Настройки (продолжение2)


Программа установки Astra Linux - Настройки
Справка: F1
Региональные настройки
Переключения языка: Ctrl+Shift
Системный язык: Русский
Часовой пояс: UTC+07:00 Россия (Новосибирск)
Дополнительные настройки: 27.11.2025, 09:12, NTP service d...
Настройки авторизации
Имя компьютера: astra-ecss
Имя администратора: eltex
Пароль: ***
Подтверждение пароля: ***
Настройки пароля GRUB: Совпадает с администраторским
Другие настройки
Компоненты ОС: Средства фильтрации сетевых паке...
Дополнительные настройки: Мандатный контроль целостности, ...
Дополнительные пакеты: Не выбраны
Ядро для установки: linux-6.1-generic (Рекомендуется)
Разметка диска: Использовать все пространство на...
[X] Перезагрузить компьютер после завершения установки
←, →, ↑, ↓, TAB - Навигация | SPACE - Изменить
ESC - Отмена | F10 - Завершение работы
Вернуться
Установить

Установка

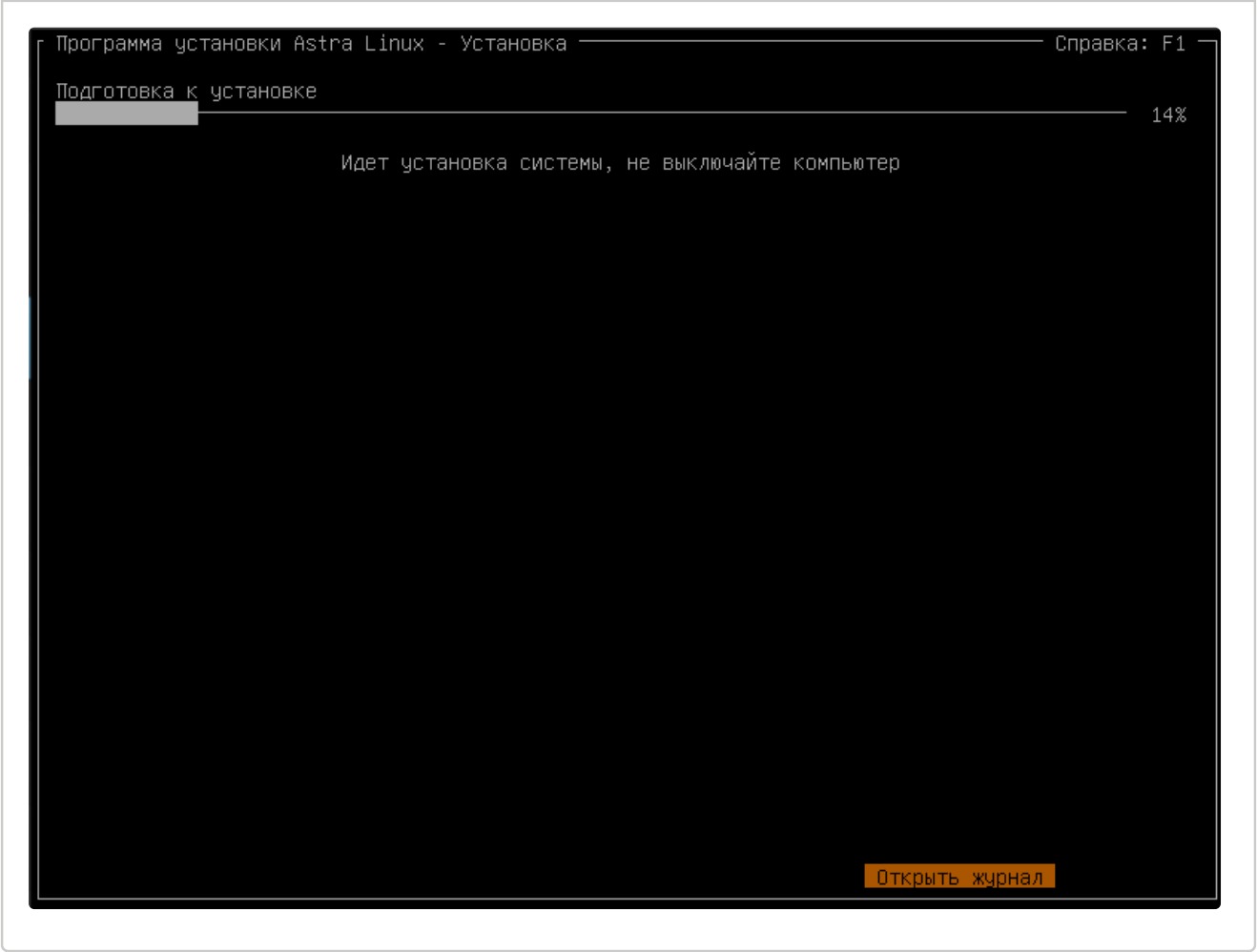


Таблица 1 — Вариант размещения информации в файловой системе на физических носителях для серверов

1	Загрузочный раздел операционной системы (создается автоматически)	boot	raid 1:hdd1,hdd2	boot	/boot	ext4	1 Gb	Первичный
2	Корневой раздел операционной системы	root	raid 1:hdd1,hdd2	root	/	ext4	30 Gb	Логический
3	Информация локальных баз данных	mnesia	raid 1:hdd1, hdd2	mnesia	/var/lib/ecss	ext4	30 Gb	Логический
4	Распределенная БД для хранения медиаресурсов	glusterfs	raid 1:hdd1, hdd2 или hdd3	glusterfs	/var/lib/ecss/restfs	ext4	Max Gb	Логический

5	Журналы функционирования подсистем ОС	log	raid 1:hdd1,hdd2 или hdd3	log	/var/log	ext4	20 Gb	Логический
6	Журналы функционирования подсистем ECSS	ecss_log	raid 1:hdd1,hdd2 или hdd3	ecss_log	/var/log/ecss	ext4	20 Gb	Логический
7	Базы данных	ecss_db	raid 1:hdd1,hdd2 или hdd3	ecss_db	/srv/ecss/ecss-postgres-bdr-ssw/	ext4	100–400 Gb*	Логический
8	Файлы пользователя	home	raid 1:hdd1,hdd2 или hdd3	home	/home	ext4	10 Gb	Логический

⚠ * Рекомендуемое значение для серий **Light, Light+, Midi** — 100 Gb. Рекомендуемое значение для серии **Heavy** — 200 Gb, **Super Heavy** — 400 Gb.

Для работы системы необходимо как **минимум** 256 Gb свободного пространства.

На серверах системы необходимо настроить параметр **"hostname".**

На всех серверах системы желательно указать одинаковое имя пользователя (любое, кроме **ssw**). Лицензия ECSS-10 привязывается к ключу eToken/ruToken и к имени компьютера (**hostname**). Системный пользователь **ssw** создается при инсталляции пакета **ecss-user**.

✔ **Рекомендуемое** значение hostname — **ecss1**;

11.1.4 Проверка установки ОС

Проверка в основном сводится к правильности создания разделов диска и наличия доступа по ssh.

Для вывода информации о состоянии дискового пространства введите команду **df -h**. Она покажет общее и занятое место в разделах. Размеры разделов должны соответствовать проекту и введенным значениям при установке.

Для проверки доступа по ssh с машины, находящейся в одной подсети со вновь установленным сервером, нужно выполнить команду:

```
ssh <user>@<IP_ecss>
```

где:

- <user> — имя пользователя, заданного при установке;
- <IP_ecss> — IP-адрес хоста, заданного при установке.

11.1.5 Настройка /etc/hosts

посмотреть можно командой ***cat /etc/hosts***
изменить/добавить командой ***sudo nano /etc/hosts***

```
sudo nano /etc/hosts
```

```
127.0.0.1 localhost
10.0.10.40 ecss1
```

11.2 Конфигурация репозиториев

После установки для обновления будет использоваться только образ DVD диска, загрузка с репозитория отключена. Давайте разрешим обновление пакетов с репозитория Astra Linux.

```
sudo nano /etc/apt/sources.list
```

привести информацию в файле к следующему виду:

```
deb https://download.astralinux.ru/astra/frozen/1.8_x86-64/1.8.3/uu/1/extended-
repository/ 1.8_x86-64 main non-free non-free-firmware contrib
deb https://download.astralinux.ru/astra/frozen/1.8_x86-64/1.8.3/uu/1/main-repository/
1.8_x86-64 main contrib non-free non-free-firmware
#deb cdrom:[OS Astra Linux 1.8.1.12  DVD]/ 1.8_x86-64 contrib main non-free non-free-
firmware
```

Добавить репозиторий Eltex:

```
sudo sh -c "echo 'deb [arch=amd64] http://archive.eltex.org/ssw/1.8_x86-64/3.18 stable main
extras external' >> /etc/apt/sources.list.d/eltex-ecss10-stable.list"
```

Далее необходимо выполнить импорт ключа командой:

```
sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 33CB2B750F8BB6A5
```

11.2.1 Изменение приоритета репозиториев

Для установки Erlang, Nodejs и Elixir из репозитория Eltex, необходимо изменить приоритет репозиториев для данных пакетов.

Добавить в файл /etc/apt/preferences.d/erlang строки:

```
sudo sh -c "echo -e 'Package: erlang*\nPin: version 2:24.3*\nPin-Priority: 1000' >> /etc/apt/
preferences.d/erlang"
```

Результат: cat /etc/apt/preferences.d/erlang

```
Package: erlang*
Pin: version 2:24.3*
Pin-Priority: 1000
```

Добавить в файл /etc/apt/preferences.d/nodejs строки:

```
sudo sh -c "echo -e 'Package: nodejs*\nPin: version 14.20.0-1nodesource1*\nPin-Priority: 1000'
>> /etc/apt/preferences.d/nodejs"
```

Результат: cat /etc/apt/preferences.d/nodejs

```
Package: nodejs*
Pin: version 14.20.0-1nodesource1*
Pin-Priority: 1000
```

Добавить в файл /etc/apt/preferences.d/elixir строки:

```
sudo sh -c "echo -e 'Package: elixir*\nPin: version 1.14*\nPin-Priority: 1000' >> /etc/apt/
preferences.d/elixir"
```

Результат: cat /etc/apt/preferences.d/elixir

```
Package: elixir*
Pin: version 1.14*
Pin-Priority: 1000
```

Выполнить обновление:

```
sudo apt update
```

Установить пакет ntp следующей командой:

на обоих хостах

```
sudo apt install -y ntp
```

11.2.2 "Оптимизация" Операционной Системы

Выставить параметры ОС в режим производительности

Используем утилиту cpufrequtils.

```
sudo apt install cpufrequtils
```

по умолчанию после инсталляции ОС использует режим "ondemand" - "по запросу" (производительность CPU по запросу приложений, экономит электроэнергию , но ниже производительность):

```
cat /etc/init.d/cpufrequtils | grep GOVERNOR=
```

в выходном сообщении системы , режим работы по умолчанию после инсталляции - "ondemand" :

```
GOVERNOR="ondemand"
```

Установить режим - результативность/производительность - в файле /etc/init.d/cpufrequtils значение "**ondemand**" заменить на "**performance**"

```
sudo sed -i 's/GOVERNOR="ondemand"/GOVERNOR="performance"/g' /etc/init.d/cpufrequtils
```

Перезапустить утилиту:

```
sudo /etc/init.d/cpufrequtils restart
```

Затем выполнить команду:

```
sudo systemctl daemon-reload
```

Отключить SWAP

Ubuntu сервер SSW работает **в реальном масштабе времени** ,
поэтому **все** необходимые **данные должны находится в оперативной памяти,** использование файла подкачки (swap-файл - /swap. img) может привести к увеличению времени обработки вызовов приложения SSW ECSS10 , **что недопустимо**. Swap - отключаем.
Выполнить последовательно три команды:

Отключение Swap:

```
sudo swapoff -a
```

Удалить файл swap.img.

```
sudo rm /swapfile
```

Удалить эту строчку (//swapfile swap swap defaults 0 0)

```
sudo nano /etc/fstab
```

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda2 during curtin installation
```

```
UUID=4c88cb4a-42c9-4b52-98c2-78da7e23b621 /                       ext4    defaults
1 1
```

Для проверки выполните команду **free -h:**

```
free -h
```

Размер Swap равен 0 - то есть он выключен

```
free -h
               total        used        free      shared  buff/cache   available
Mem:           3,9G        110M        3,2G        820K        535M        3,5G
Swap:            0B          0B          0B
```

Установка часового пояса

При инсталляции astra linux **не предлагается** установить часовой пояс(по умолчанию устанавливается время UTC). Его нужно установить **вручную** (для корректной работы системы тарификации , работ по расписанию и т.д.), например:

```
sudo timedatectl set-timezone Asia/Novosibirsk
```

Улучшение работы высоконагруженных серверов

Улучшить работу высоконагруженных серверов можно увеличив лимит открытых файлов.

Для установки лимита открытых файлов необходимо:

Проверить текущий лимит командой:

```
ulimit -a
```

результат :

```
eltex@ecss1:~$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
scheduling priority             (-e) 0
file size               (blocks, -f) unlimited
pending signals                 (-i) 15515
max locked memory       (kbytes, -l) 65536
max memory size         (kbytes, -m) unlimited
open files                      (-n) 1024
pipe size            (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
real-time priority              (-r) 0
stack size              (kbytes, -s) 8192
cpu time               (seconds, -t) unlimited
```

```
max user processes              (-u) 15515
virtual memory          (kbytes, -v) unlimited
file locks                      (-x) unlimited
```

Данного лимита (open files **1024**) **недостаточно** для нормальной работы высоконагруженных серверов.

Установить лимит открытых файл для каждого пользователя:

```
sudo sed -i  '55i\*               soft    nproc           65536\n*                hard
nproc           131072\n*               soft    nofile          65536\n*                hard
nofile          131072\nroot            -       memlock         unlimited' /etc/security/
limits.conf
```

Результат:

```
# /etc/security/limits.conf
#
#Each line describes a limit for a user in the form:
#
#<domain>        <type>  <item>  <value>
#
#Where:
#<domain> can be:
#        - a user name
#        - a group name, with @group syntax
#        - the wildcard *, for default entry
#        - the wildcard %, can be also used with %group syntax,
#                 for maxlogin limit
#        - NOTE: group and wildcard limits are not applied to root.
#          To apply a limit to the root user, <domain> must be
#          the literal username root.
#
#<type> can have the two values:
#        - "soft" for enforcing the soft limits
#        - "hard" for enforcing hard limits
#
#<item> can be one of the following:
#        - core - limits the core file size (KB)
#        - data - max data size (KB)
#        - fsize - maximum filesize (KB)
#        - memlock - max locked-in-memory address space (KB)
```

```
#        - nofile - max number of open files
#        - rss - max resident set size (KB)
#        - stack - max stack size (KB)
#        - cpu - max CPU time (MIN)
#        - nproc - max number of processes
#        - as - address space limit (KB)
#        - maxlogins - max number of logins for this user
#        - maxsyslogins - max number of logins on the system
#        - priority - the priority to run user process with
#        - locks - max number of file locks the user can hold
#        - sigpending - max number of pending signals
#        - msgqueue - max memory used by POSIX message queues (bytes)
#        - nice - max nice priority allowed to raise to values: [-20, 19]
#        - rtprio - max realtime priority
#        - chroot - change root to directory (Debian-specific)
#
#<domain>      <type>  <item>         <value>
#

#*               soft    core            0
#root             hard    core            100000
#*               hard    rss             10000
#@student         hard    nproc           20
#@faculty         soft    nproc           20
#@faculty         hard    nproc           50
#ftp              hard    nproc           0
#ftp              -       chroot          /ftp
#@student         -       maxlogins       4
*                soft    nproc           65536
*                hard    nproc           131072
*                soft    nofile          65536
*                hard    nofile          131072
root             -       memlock         unlimited

# End of file
```

11.3 Настройка сетевых интерфейсов

⚠ Получение адресов на сетевых интерфейсах по DHCP **недопустимо** на серверах ECSS.

Сетевые настройки необходимо выполнять с помощью Netplan.

По умолчанию в Astra Linux пакет netplan не установлен.

Установить пакеты для обслуживание функции netplan:

```
sudo apt install -y dialog libssl1.0.0 netplan.io
```

Сейчас можно выполнить конфигурацию netplan:

```
sudo nano /etc/netplan/ecss.yaml
```

11.3.1 Пример netplan

Netplan для интерфейсов сервера ecss1 (/etc/netplan/ecss.yaml)

на ecss1

```
# Netplan для хоста ecss1 программного коммутатора
# Обратите внимание на обязательное наличие минимум двух пробелов в каждой строке и секции
 (кроме строки секции network)

network:
  version: 2 # Версия netplan
  renderer: networkd # Исполнитель конфигураций netplan
  ethernets: # Раздел описания интерфейсов ethernet
    enp0s3: # Название интерфейса ВМ к интернет
        dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
        dhcp6: no
        addresses: [192.168.56.40/24]
    enp0s8: # Название интерфейса для SSW
        dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
        dhcp6: no
  vlans:
    net.10: # Интерфейс управления
        id: 10
        link: enp0s3
        addresses: [10.0.10.40/24]
    net.20: # Интерфейс для VoIP
        id: 20
        link: enp0s8
        addresses: [10.0.20.40/24]
```

Так же проверить отсутствие в каталоге **/etc/netplan/** других файлов, если другие файлы присутствуют , то их нужно переместить в другой каталог или удалить, в противном случае возможна некорректная настройка сетевых интерфейсов и некорректная работа SSW.

```
ls -la /etc/netplan/
```

Применить, установленные параметры командой:

```
sudo netplan apply
```

Просмотреть получившиеся настройки можно при помощи команд **ifconfig** или **ip a**:

Для применения новых сетевых настроек необходимо выполнить команду **netplan apply.** Перезапуск сети или системы не требуется. Подробнее про настройки netplan см. в Приложении E. Netplan.

11.4 Обновление ОС и инсталляция необходимого ПО

11.4.1 Обновление системы

Перед началом установки необходимо обновить ОС:

```
sudo apt update
```

```
sudo apt upgrade --enable-upgrade
```

11.4.2 Инсталляция необходимого ПО

Установка дополнительных пакетов программ

Для работы SSW, потребуются дополнительные пакеты ПО не установленные в Astra Linux по умолчанию, но доступные в репозитории.

```
sudo apt install -y ntp iptables tcpdump vlan dnsmasq aptitude atop ethtool htop mc screen ssh
 tftpd-hpa sngrep tshark gnuplot debconf-utils ncdu wget
```

Пакеты ПО которые рекомендуется установить для работы

Список **обязательного** сервисного программного обеспечения:

```
sudo apt install ntp tcpdump vlan dnsmasq
```

ntp	NTP-сервер

tcpdump	сниффер пакетов
vlan	управление VLAN
dnsmasq	легковесный DNS/DHCP-сервер

Список **рекомендуемого** диагностического и вспомогательного программного обеспечения:

```
sudo apt install aptitude atop ethtool htop mc screen ssh tftpd sngrep tshark gnuplot
debconf-utils iotop ncdu
```

aptitude	установка программ из репозиториев, рекомендуется использовать вместо программы apt/apt-get
atop	мониторинг загрузки хоста с функцией периодического сохранения информации в файлы
ethtool	просмотр статистики сетевых интерфейсов
htop	мониторинг процессов
mc	файловый менеджер
screen	мультиплексор терминалов
ssh	сервер и клиент SSH
tftpd	TFTP-сервер
sngrep	трассировка sip
tshark	консольный аналог wireshark
gnuplot	вывод графиков статистики
debconf-utils	набор утилит для работы с базой debconf
iotop	инструмент для мониторинга использования ввода-вывода (IO) на диске в реальном времени в Linux
ncdu	утилита для поиска больших директорий в системе Linux

Для **просмотра** установленных пакетов выполнить следующую команду (**данный пункт является необязательным**: его можно выполнить, если не уверены, что какие-то пакеты установлены):

на обоих хостах

```
sudo dpkg --get-selections
```

⚠ Перед началом установки пакетов ecss , нужно убедится в соответствии полосы пропускания СПД необходимым требованиям.
Для этого выполнить команду **sudo ethtool <имя интерфейса>** для всех физических интерфейсов.
Пример:

```
sudo ethtool net.20
Settings for net.20:
    Supported ports: [ TP ]
    Supported link modes:   10baseT/Half 10baseT/Full
                            100baseT/Half 100baseT/Full
                            1000baseT/Full
    Supported pause frame use: No
    Supports auto-negotiation: Yes
    Supported FEC modes: Not reported
    Advertised link modes:  10baseT/Half 10baseT/Full
                            100baseT/Half 100baseT/Full
                            1000baseT/Full
    Advertised pause frame use: No
    Advertised auto-negotiation: Yes
    Advertised FEC modes: Not reported
    Speed: 1000Mb/s
    Duplex: Full
    Auto-negotiation: on
    Port: Twisted Pair
    PHYAD: 0
    Transceiver: internal
    MDI-X: off (auto)
    Link detected: yes
```

Проверить значение следующих параметров:
Advertised auto-negotiation: **Yes**
Speed: **1000Mb/s** (не менее)
Duplex: **Full**

11.5 Инсталляция пакетов ECSS

11.5.1 Предварительные требования

- Установленная и обновленная операционная система Astra Linux;
- Отсутствие в системе пользователя с именем **ssw**;
- Разбиение дискового пространства в соответствии с рекомендациями;
- Настроенная сеть;
- Установленный набор необходимых пакетов;

- Доступ к репозиторию ELTEX.

установка пакетов deb

```
   ┌ версия ПО ECSS. Сейчас в заявке на сертификацию она заявлена как 3.

   │ Мы её менять не будем до следующей сертификации или
  │ выпуска принципиально новой версии системы.

   │   ┌ версия System Release. Общая версия для всех компонент входящих в конкретный релиз.

   │      │ Меняем её централизовано,
   │   │ принимаем решение о выпуске нового релиза. Релизы между собой обычно не совместимы.
┌──┘ ┌──┘
ECSS.SysRel.SubMaj.SubMin
              └────┘ └────┘
        │   └ версия Subsystem minor. Минорную версию подсистемы устанавливает разработчик
подсистемы.

                  │          Минорную версию
        │     меняем при добавлении патча. Как правило, минорные версии в рамках одной мажорной
версии
        │     совместимы между собой и отличаются в рамках конкретных патчей.

                  └ версия Subsystem major. Мажорная версия устанавливается разработчиком
подсистемы.

                     Мажорную версию подсистемы необходимо менять при внесении в подсистему
существенных изменений.
```

Утилита установки пакетов APT анализирует версию пакета слева направо, то есть, для примера, мы имеем пакет 14.14.7.7 ,
в репозитории находятся пакеты
14.14.7.8
14.14.7.9
14.14.8.1
14.14.20 -14.14.28
то при выполнении команды **sudo apt install имя пакета** → будет автоматически проверен и установлен пакет 14.14.28 , так как он самый последний, анализ будет выполнен по 3-й позиции (major), анализ по 4-й (minor) выполнен не будет (аналогичным образом будет выполнятся команда **sudo apt upgrade --enable-upgrade**).
В случае если для конкретной ситуации требуется перейти с версии 14.14.7.7 на версию 14.14.7.9 , стандартная команда → **sudo apt upgrade --enable-upgrade** , нам не поможет, так как будет выбран самый новый пакет, в данной ситуации нам **необходимо в явном виде** указать какую версию пакета мы хотим установить, в данном примере мы **должны** выбрать команду → **sudo apt install имя пакета=14.14.7.9** . Обычно это необходимо для тестирования определенного патча, для стандартных обновлений достаточно выбора привычной команды установки/обновления пакета

В ходе установки пакетов ECSS нужно будет ответить на ряд вопросов для формирования необходимой конфигурации. Для автоматической загрузки требуемых настроек можно воспользоваться командами из пакета debconf-utils.
Описание и примеры использования при работе с debconf приведены в Приложении B. Debconf.

Для инсталляции системы ECSS-10 необходимо устанавливать пакеты в порядке, в котором они описаны ниже в документации.

11.5.2 Инсталляция обязательных пакетов

Установка пакета ecss-dns-env

Выполнить следующую команду:

на обоих хостах

```
sudo apt install -y ecss-dns-env
```

Настройщик предложит выбрать разделы для настройки по вопросам ниже. **Не выбираем ничего** просто нажать **Enter**.

Настройка преобразование DNS имени в IP адрес

Для корректной работы системы, после установки пакета ecss-node требуется скорректировать дефолтные установки . Для этого:

В файле /etc/resolv.conf изменить параметр nameserver установленный системой по умолчанию на 127.0.0.1 (если строк nameserver несколько - остальные удалить/закомментировать):

```
sudo nano /etc/resolv.conf
```

Пример:

```
domain eltex.loc
search eltex.loc
nameserver 127.0.0.1
```

В Astra Linux после перезагрузки файл /etc/resolv.conf приводится к исходному виду, поэтому его **нужно сделать неизменяемым**:

```
sudo chattr +i /etc/resolv.conf
```

Для доступа к репозиториям AstraLinux и Элтекс нужно указать локальный DNS или глобальный (для примера 8.8.8.8 или 77.88.8.8)

Добавить в файл /etc/dnsmasq.d/ecss строки server=<Ipadd DNS сервера> (в примере 77.88.8.8) :

```
sudo sh -c "echo 'server=77.88.8.8' >> /etc/dnsmasq.d/ecss"
```

Перезапустить сервис:

```
sudo systemctl restart dnsmasq.service
```

Установка Postgres

Примечание

Перед установкой `ecss-postgres-bdr-ssw` в системе, в `apt sources,` должен быть добавлен `docker для этого выполнить следующие команды:`

```
sudo install -m 0755 -d /etc/apt/keyrings
```

```
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/
keyrings/docker.asc
```

```
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

```
sudo sh -c "echo 'deb [arch=amd64 signed-by=/etc/apt/keyrings/docker.asc]
https://download.docker.com/linux/ubuntu   jammy stable' >> /etc/apt/
sources.list.d/docker.list"
```

Выполнить обновление :

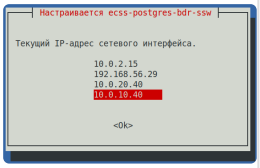
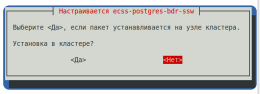
```
sudo apt update && sudo apt upgrade --enable-upgrade
```

Установка пакета ecss-postgres-bdr-ssw

ecss1

```
sudo apt install -y ecss-postgres-bdr-ssw
```

При конфигурации пакета **ecss-postgres-bdr-ssw** будут заданы вопросы:

Вопросы ecss-postgres-bdr-ssw	Ответы для ecss1 (Master)	Пример
Текущий IP-адрес сетевого интерфейса.	**Выбрать** из списка IP-адрес для работы с Postgres	
Установка в кластере?	**No** (необходимо выбрать)	

Результат:

```
============================================================
============================================================
 ecss-postgres-bdr-ssw successfully installed/updated!
============================================================
============================================================
```

Установка ecss-node

Установка обязательного пакета **ecss-node** включает в себя установку и первоначальную настройку основных подсистем.

⚠ В системе уже должен быть установлен пакет **ecss-postgres-bdr-ssw**

Для установки пакета **ecss-node** выполните команду:

первоначально на ecss1

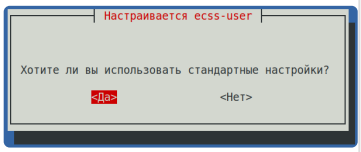
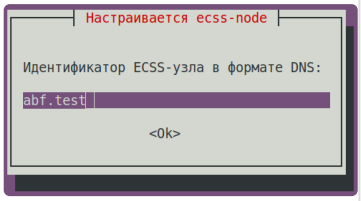
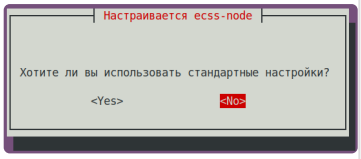
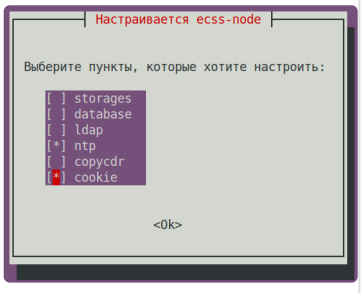
```
sudo apt install -y ecss-node
```

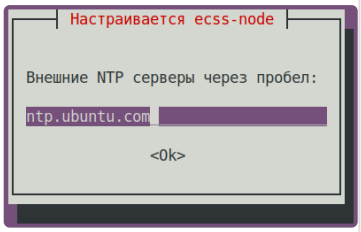
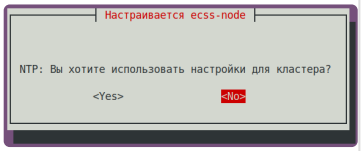
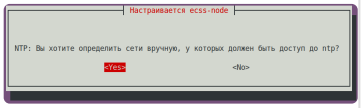
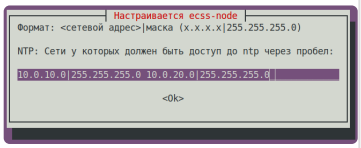
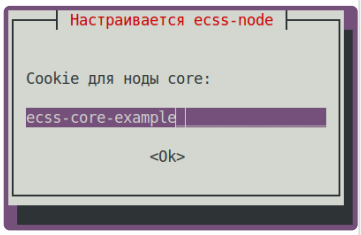
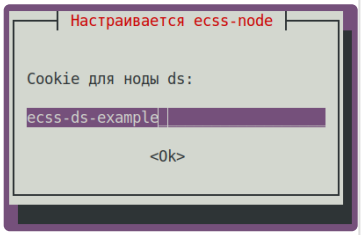
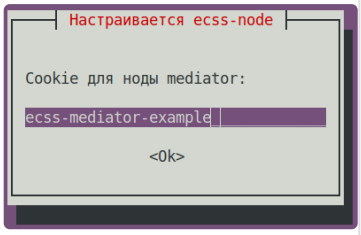
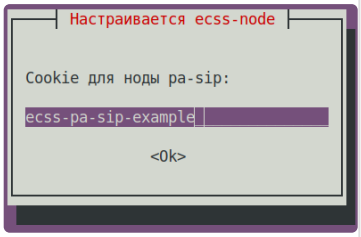
Во время установки пакета создается пользователь **ssw**, от имени которого запускаются все сервисы ecss*. Создаются необходимые каталоги, выполняется настройка DNS, идет настройка SSL-сертификатов.

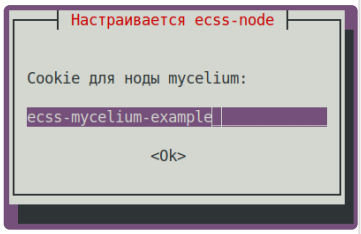
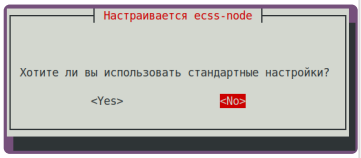
✔ При установки так же будет установлен пакет **ecss-user**

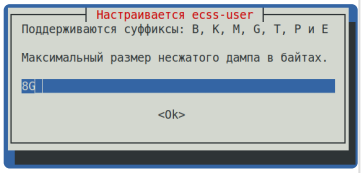
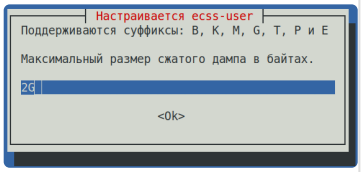
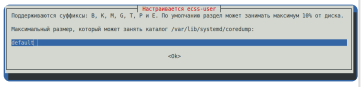
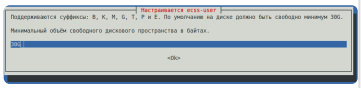
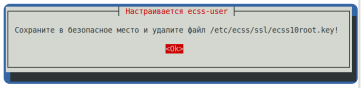
Установка на сервере **ecss1**

Во время установки будет предложено настроить параметры, необходимые для формирования конфигурационных файлов, примеры ответов следует ниже.

Вопросы ecss-user	**Ответы для ecss1**	**Пример**
Хотите ли вы использовать стандартные настройки?	Yes (значение по умолчанию)	
Вопросы ecss-node	**Ответы для ecss1**	**Пример**
Идентификатор ECSS-узла в формате DNS (Настройка имени кластера): (любое уникальное имя, в примере abf.test)	**abf.test** (необходимо ввести) Формат **ECSS_ID** Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире. `[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]`	
Хотите ли вы использовать стандартные настройки?	**No** (необходимо выбрать)	
Выберите пункты, которые хотите настроить:	**ntp + cookie** (необходимо выбрать)	

Внешние NTP серверы через пробел:	ntp.ubuntu.com (значение по умолчанию)	
NTP: Вы хотите использовать настройки для кластера?	**No** (необходимо выбрать)	
NTP: Вы хотите определить сети вручную, у которых должен быть доступ до ntp?	**No** (необходимо выбрать)Yes (значение по умолчанию)	
NTP: Сети у которых должен быть доступ до ntp через пробел: (в примере 10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0)	**10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0** (необходимо ввести)	
Cookie для ноды core: (указать уникальный cookie для core , в примере ecss-core-example)	**ecss-core-example** (необходимо ввести)	
Cookie для ноды ds: (указать уникальный cookie для ds , в примере ecss-ds-example)	**ecss-ds-example** (необходимо ввести)	
Cookie для ноды mediator: (указать уникальный cookie для mediator , в примере ecss-mediator-example)	**ecss-mediator-example** (необходимо ввести)	
Cookie для ноды pa-sip: (указать уникальный cookie для pa-sip , в примере ecss-pa-sip-example)	**ecss-pa-sip-example** (необходимо ввести)	

Cookie для ноды mycelium: (указать уникальный cookie для mycelium , в примере ecss-mycelium-example)	**ecss-mycelium-example** (необходимо ввести)	
Cookie для ноды sorm: (указать уникальный cookie для sorm , в примере ecss-sorm-example)	**ecss-sorm-example** (необходимо ввести)	

Вопросы ecss-user	**Ответы для ecss1**	**Пример**
Максимальный размер несжатого дампа в байтах.	8G (значение по умолчанию)	
Максимальный размер сжатого дампа в байтах.	2G (значение по умолчанию)	
Максимальный размер, который может занять каталог /var/lib/systemd/coredump:	default (значение по умолчанию)	
Минимальный объем свободного дискового пространства в байтах.	30G (значение по умолчанию)	
Сохраните в безопасное место и удалите файл /etc/ecss/ssl/ecss10root.key!	Ok (значение по умолчанию)	

Проверить статус сервисов следующей командой:

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
```

В случае статуса "active" продолжаем дальше, в случае статуса "inactive" выполнить команду "sudo systemctl restart ecss-<имя пакета>"

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
active
active
active
```

```
active
active
```

Установка лицензии

Система ECSS-10 использует ELM-лицензирование – сервис распределения лицензий ECSS-экосистемы, состоит из **ecss-license-agent** и **ecss-license-master** .

- **ECSS License Agent** - Адаптер для работы с EcssLicenseMaster (LM). Получает список хостов EcssLicenseMaster, реализует логику подключения и переподключения к LM.
  - Отвечает за получение, валидацию и дальнейшую отправку лицензий клиенту.
  - Отправляет статусы подключения и полученные лицензии клиенту.
  - Предоставляет API для отправки HTTP запросов и WS событий на LM.
- **Ecss License Master** - Сервис оркестрации параметров лицензии для ECSS-узлов.
  - Отвечает за:
    - Получение лицензии для конкретного ECSS-узла (SSW / AuP) либо устройства (SMG)
    - Распределение параметров лицензии между ECSS-узлами и ECSS-сервисами
    - Контроль активности ECSS-узлов и их валидация во избежание тиражирования лицензии сверх указанных лимитов
    - Контроль изменения указанных лимитов на ECSS-узлах, и их динамическое распределение в соответствии с текущей нагрузкой.

ecss-license-agent является составной частью SSW, входит в пакет ecss-node. Отдельной конфигурации не требует.

В свою очередь **Ecss License Master** может быть установлен как на SSW так и на отдельно выделенный сервер. В случае если Ecss License Master уже установлен на вашем проекте, то потребуется только дополнить запись о новом компоненте в конфигурационном файле LM, и следующий пункт – установка "Установка LM" не потребуется.

Установка LM на SSW host

Для установки пакета **ecss-license-master** необходимо добавить репозиторий следующей командой:

```
sudo sh -c "echo deb [arch=amd64] https://archive.eltex.org/ssw/1.8_x86-64/aup/2.0 stable
main extras external >> /etc/apt/sources.list.d/lm-stable.list"
```

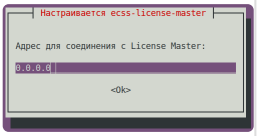
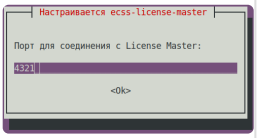
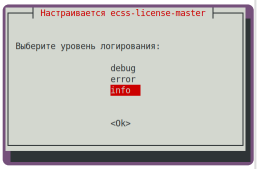
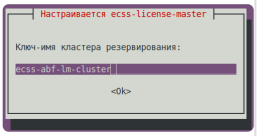
Выполнить обновление из добавленного репозитория командой:

```
sudo apt update
```

Установить пакет LM (ecss-license-master)

```
sudo apt install ecss-license-master
```

При установке пакета ecss-license-master будут заданы вопросы:

Вопросы	Ответы	Примеры
Адрес для соединения с License Master: License Master listen ip:	0.0.0.0 (значение по умолчанию)	
Порт для соединения с License Master: License Master listen port:	4321 (значение по умолчанию)	
Выберите уровень логирования: Log level:	info (значение по умолчанию), возможные варианты: debug, error, info	
Ключ-имя кластера резервирования: License Master cluster key:	указать уникальное имя ecss-lm-cluster (значение по умолчанию) Менеджеры с одинаковыми именами попытаются собраться в кластер и резервировать друг друга.	

Данные сохраняются в файле - **/etc/ecss/ecss-license-master/config.env** :

```
ECSS_LM_HOST=0.0.0.0
ECSS_LM_PORT=4321
ECSS_LM_LOG_LEVEL=info
RELEASE_COOKIE=ecss-abf-lm-cluster
```

Конфигурация LM

Сервис ecss-license-master использует два конфигурационных файла: **/etc/ecss/ecss-license-master/config.env** и **/etc/ecss/ecss-license-master/config.yaml**.
В файл **config.env** информация сохраняется в момент установки пакета **ecss-license-master**.
Config.yaml файл содержит основные настройки для подключения к ELM серверу, откуда брать лицензии, какие лицензии, и куда их потом передавать, **все это нужно сконфигурировать вручную**.

При установке LM на SSW

Выполнить команду:

```
sudo nano /etc/ecss/ecss-license-master/config.yaml
```

информация в файле после установки

```
elm_addresses: []
licenses: []
ecss_nodes: []
```

В поле **elm_adresses** нужно прописать список серверов ELM к которым мы хотим подключаться для получения лицензий. Первый адрес в списке используется как основной. Дополнительные строки с адресами работают как резерв.

Данные ELM сервера(ов) получить у менеджера проекта.

пример:
- "elm-1.eltex.loc:8099"
- "192.168.111.22:8099"
- "elm-3.eltex.loc:8099"

Дальше задаются сами лицензии в поле **licenses**.

- **id** – может быть любой, главное чтоб был уникальный у каждой лицензии. Влияет только на адрес лицензии на самом LM (например 0,1,2,3).
- **kind** – для SSW поддерживается только ***ssw*.**
- **type** – выбирается как ***elm*** если запрашивать данную лицензию нужно с ELM сервера. Если нужно использовать файловую лицензию SSW, то тип выбирается как ***ecss_license***.
  - при ***elm*** типе, необходимо задать поля **license_key** и **product_id** для идентификации лицензии.
  - при ***ecss_license*** типе, задается поле **license** содержащее ключ статичной лицензии

Параметры **license_key** и **product_id** должны быть согласованы с данными загруженными на ELM сервере .

Значение параметров **license_key**, **product_id** или **license**, **passport** нужно получить у менеджера проекта.

Блок **ecss_nodes**.
В нем нужно задать данные сервисов которые будут получать лицензию. В нашем случае SSW.
Сервис задается тремя параметрами:

- **id –** идентификатор сервиса, должен совпадать с ECSS_ID лицензируемого SSW.
- **kind –** тип лицензируемого сервиса. по аналогии с блоком лицензий, для SSW поддерживается только ***ssw*.**
- **license_id –** идентификатор лицензии которая будет отправляться на данный сервис. Соответствует идентификаторам лицензии из блока **licenses**

Два SSW с **разными ECSS_ID** не могут получить одну и ту же лицензию, конфигурация с таким распределением будет помечена некорректной.
Кроме того, два SSW с **одинаковыми ECSS_ID** не смогут одновременно получать одну и ту же лицензию с одного LM,
поэтому крайне важно корректно устанавливать свой ECSS_ID и делать его уникальным как минимум в зоне видимости используемого LM.

Формат **ECSS_ID**
Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире.

```
[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]
```

Для SSW работающего в кластере указываем только одну запись в блок **ecss_nodes** на кластер, индивидуально для каждого хоста указывать нельзя. Иначе будут генерироваться два запроса с одинаковым ECSS_ID что недопустимо.

Пример: файла config.yaml

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
   kind: SSW
   type: elm
   license_key: "ssw1test"
   product_id: "ECSS1000001"
 - id: 1
   kind: ssw
   type: ecss_license
   license: "af615ebb92d381125ff"
ecss_nodes:
 - id: abf.test
   kind: ssw
   license_id: 0
```

Выполнить перезагрузку сервиса **ecss-license-master.service** для обновления конфигурации из файла **/etc/ecss/ecss-license-master/config.yaml** следующей командой:

если кластер, то на обоих хостах

```
sudo systemctl restart ecss-license-master.service
```

При подключении SSW к ранее установленному LM

В случае если LM уже установлен и используется в локальной сети требуется добавить параметры нового SSW в конфигурационный файл **/etc/ecss/ecss-license-master/config.yaml** в блоках licenses и ecss_nodes.

Выполнить команду:

если кластер,то на обоих хостах

```
sudo nano /etc/ecss/ecss-license-master/config.yaml
```

для примера:

Было:

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
   kind: SSW
   type: elm
   license_key: "ssw1test"
   product_id: "ECSS1000001"
 - id: 1
   kind: ssw
   type: ecss_license
   license: "af615ebb92d381125ff"
 - id: 2
   kind: SSW
   type: elm
   license_key: "ssw2test"
   product_id: "ECSS1000002"
ecss_nodes:
 - id: 1.test
   kind: ssw
   license_id: 0
 - id: 2.test
   kind: ssw
   license_id: 1
 - id: 3.test
   kind: ssw
   license_id: 2
```

Стало:

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
   kind: SSW
   type: elm
   license_key: "ssw1test"
   product_id: "ECSS1000001"
 - id: 1
   kind: ssw
```

```
    type: ecss_license
    license: "af615ebb92d381125ff"
  - id: 2
    kind: SSW
    type: elm
    license_key: "ssw2test"
    product_id: "ECSS1000002"
  - id: 3
    kind: SSW
    type: elm
    license_key: "ssw3test"
    product_id: "ECSS1000003"
ecss_nodes:
  - id: 1.test
    kind: ssw
    license_id: 0
  - id: 2.test
    kind: ssw
    license_id: 1
  - id: 3.test
    kind: ssw
    license_id: 2
  - id: abf.test
    kind: ssw
    license_id: 3
```

Выполнить перезагрузку сервиса **ecss-license-master.service** для обновления конфигурации из файла **/etc/ecss/ecss-license-master/config.yaml** следующей командой:

```
sudo systemctl restart ecss-license-master.service
```

Подключение SSW к Менеджера Лицензий

Выполнить конфигурацию подключения SSW к LM следующей командой в CoCon, (в примере LM , IPadd 10.0.10.41. значение порта из файла /etc/ecss/ecss-license-master/config.env по умолчанию 4321):

выполняется в CoCon

```
/system/licence/manager/set --hosts [https://10.0.10.41:4321]
```

В случае установки LM на SSW указывать IP адреса OAM

После выполнения проверить статус подключения командой:

выполняется в CoCon

```
/system/licence/manager/show-status
┌──────────────────────┬───────┬─────┐
│         Host         │Current│Alive│
├──────────────────────┼───────┼─────┤
```

```
|https://10.0.10.41:4321|*      |true |
└──────────────────────┴───────┴─────┘
```

Должны иметь статус alive=true, и быть "current".

- Current - отображает к какому хосту подключён SSW. Если есть подключение, указывается *, если нет - ничего не указано.
- Alive - отображает статус доступности хоста (healthcheck).

Подключение к License Manager

Отсутствие подключения к License Manager \ ELM будет равноценно отсутствию доступа к токену, что приведёт к критическим авариям, переходу в аварийный режим по истечению которого, если связь не будет восстановлена, система перейдёт на дефолтную лицензию.

данное поведение применимо только при работе в **elm** режиме

Если статус к LM показан current, можно посылать запрос на загрузку лицензии на SSW командой:

выполняется в CoCon

```
/cluster/storage/ds1/licence/request
```

пример выполнения команды /cluster/storage/ds1/licence/request

type=elm

```
/cluster/storage/ds1/licence/request
Licence received
[*******                                                    ] 6s 2ms
Success: Licence parameters applied
```

В режиме "type=ecss_license" команда **/cluster/storage/ds1/licence/request** выполнится только в случае если был предварительно загружен паспорт.

type=ecss_license

```
/cluster/storage/ds1/licence/request
Waiting for licence...
[*******                                                    ] 6s
┌─┬──────────────────────────────────────────────────┬─────────┬──────────────────────────────
──────┐
|A|                   Description                    |Old Value|               New Value
      |
├─┼──────────────────────────────────────────────────┼─────────┼──────────────────────────────
──────┤
|^|Support for reservation of call-processes        |false    |true
      |
|^|Maximum call duration (in seconds)                |60       |2678400
      |
|^|Maximum number of simultaneous calls              |5        |10000
      |
```

```
|^|Elph                                                  |         |
                |
|^| total count of members of the one Elph group         |10       |100
                |
|^|Subscribers limit                                     |10       |500000
                |
|^|Virtual subscribers limit                             |infinity |1000
                |
|^|Add-on conferences                                    |         |
                |
|^| total count                                          |30       |1000
                |
|^|Add-on conferences                                    |         |
                |
|^| members of the one add-on conference                 |16       |300
                |
|^|Chat rooms                                            |         |
                |
|^| total count                                          |30       |100
                |
|^|Chat rooms                                            |         |
                |
|^| members of the one chatroom                          |16       |300
                |
|^|Meet Me                                               |         |
                |
|^| total count                                          |0        |100
                |
|^|Meet Me                                               |         |
                |
|^| members of the one Meet Me conference                |0        |300
                |
|^|SORM                                                  |         |
                |
|^| enabled                                              |false    |true
                |
|+| channels on SORM mediator                            |         |64
                |
|^|Sorm extractor                                        |none     |mfi
                |
|^|Support antifraud system                              |none     |custom
                |
|+|Call center                                           |         |
                |
|+| operator's capability for look at calls in queue     |         |true
                |
|+| operator's capability for use Intervension SS        |         |true
                |
|^|Call center                                           |         |
                |
|^| active agents                                        |0        |1000
                |
|^|Call center                                           |         |
                |
|^| active supervisors                                   |0        |100
                |
|^|Teleconference                                        |         |
                |
```

```
|^| channels                                              |3        |2000
              |
|^|Teleconference                                         |         |
              |
|^| max members of one teleconference                     |0        |200
              |
|^|Teleconference                                         |         |
              |
|^| active count                                          |0        |32
              |
|^|TSMN system                                            |         |
              |
|^| concurrent calls (active)                             |0        |50
              |
|^|TSMN system                                            |         |
              |
|^| concurrent calls (backup)                             |0        |50
              |
|^|TSMN system                                            |         |
              |
|^| backup mode                                           |none     |"backup"
              |
|^|Total count of simultaneous records voice calls        |0        |200
              |
|+|IVR                                                    |         |
              |
|+| customization enabled                                 |         |true
              |
|+| Automatic Speech Recognition (ASR) subsystem enabled  |         |true
              |
|^|Channels on dialer outgoing calls                      |2        |5
              |
|+|Supplementary Services licence package(s)              |         |
              |
|+| name                                                  |         |'ECSS-ADV'
              |
|+| limit                                                 |         |10000
              |
|+| description                                           |         |"Additional services of 4
and 5 levels" |
|+| SS list                                               |         |[1,2,3,4,5]
              |
|+| name                                                  |         |'ECSS-BAS'
              |
|+| limit                                                 |         |10000
              |
|+| description                                           |         |"Basic services of 1 and
2 levels"     |
|+| SS list                                               |         |[1,2]
              |
|+| name                                                  |         |'ECSS-BAS+'
              |
|+| limit                                                 |         |10000
              |
|+| description                                           |         |"Expansion of basic
services of 3 level"|
|+| SS list                                               |         |[1,2,3]
              |
```

```
|^|GEO backup                                                    |         |
                |
|^| enabled                                                      |false    |true
                |
|^|SIGTRAN                                                       |         |
                |
|^| SEP                                                          |false    |true
                |
|^| STP                                                          |false    |true
                |
└─┴──────────────────────────────────────────────────────────────┴─────────┴──────────────────────────
──────────────┘
Legend:
    '^' - Changed;
    '-' - Deleted;
    '+' - New.


[request] You are trying to add a license that is different from the current one by the pa
          rameters in the table (other values will remain unchanged).
Licence received      o ?> yes
[*******                                                                ] 1m 15s
Success: Licence parameters applied
```

В зависимости от выбранного типа лицензии для SSW : "type=**elm**" или "type=**ecss_license**" на команду просмотра лицензий получим разный вывод:

type=ecss_license

```
/cluster/storage/ds1/licence/list-licence
┌─┬──────────────────┬──────────┬──────┬──────────────────┬──────────────────────┬──────────
──────────┬──────────┐
|Id|Creation date(UTC) |   SSW ID   |Active|  Description   |Comm. Exp.
Date(UTC)|Expiration date(UTC)|Time left|
├─┼──────────────────┼──────────┼──────┼──────────────────┼──────────────────────┼──────────
──────────┼──────────┤
|1 |12.03.2026 18:40:00|ECSS 010070 |*     |ECSS TPM License|13.03.2027 06:59:59
|14.03.2026 10:44:05 |1d 6h 59m|
|0 |01.01.1990 00:00:00|ECSS DEFAULT|      |Default licence |                      |
|         |
└─┴──────────────────┴──────────┴──────┴──────────────────┴──────────────────────┴──────────
──────────┴──────────┘
```

TPM - является полным аналогом лицензирования которое использовалось раньше, с токеном и паспортом, без подключения к LM.

или

type=elm

```
/cluster/storage/ds1/licence/list-licence
┌──┬────────────────────┬──────────────┬──────┬──────────────────┬──────────────────────┬──────────────────┬──────────┐
│Id│Creation date(UTC)  │   SSW ID     │Active│  Description     │Comm. Exp. Date(UTC)  │Expiration date(UTC)│Time left│
├──┼────────────────────┼──────────────┼──────┼──────────────────┼──────────────────────┼──────────────────┼──────────┤
│1 │03.02.2026 16:26:34 │ECSS2000009   │*     │ECSS ELM License  │03.02.2027 16:26:34   │14.03.2026 10:49:12 │1d 6h 59m│
│0 │01.01.1990 00:00:00 │ECSS DEFAULT  │      │Default licence   │                      │                  │          │
└──┴────────────────────┴──────────────┴──────┴──────────────────┴──────────────────────┴──────────────────┴──────────┘
```

Установка остальных пакетов ecss и их настройка

Выполнить конфигурацию для Sip адаптера , выполнив следующую команду:

на обоих хостах

```
sudo systemctl edit ecss-pa-sip.service
```

Добавить в конфигурационный файл следующие данные:

```
[Service]
LimitNOFILE=65536
```

Перезагрузите конфигурацию

на обоих хостах

```
sudo systemctl daemon-reload
```

Актуализировать конфигурационный файл, выполнив перезагрузку сервера:

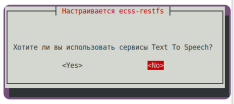
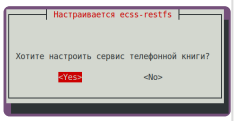
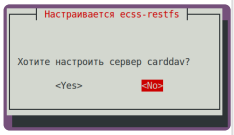
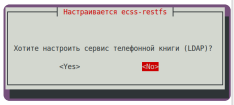
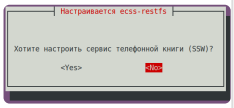
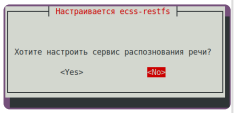
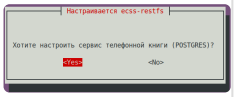
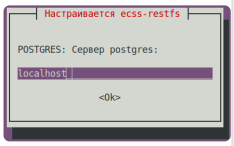
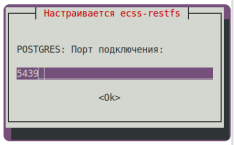
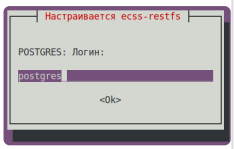
```
sudo reboot
```

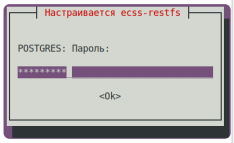
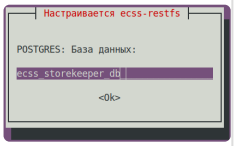
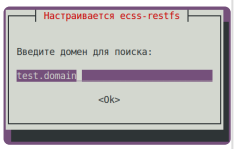
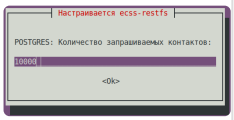
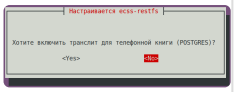
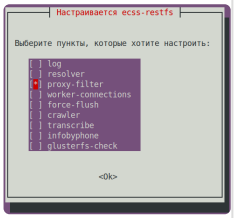
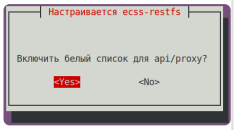
Установка пакета ecss-restfs и его настройка

установить на ecss1, затем на ecss2

```
sudo apt install -y ecss-restfs
```

Установка **ecss-restfs.** В ходе установки нужно будет ответить на ряд вопросов для создания необходимых конфигурационных файлов. Также инсталлятор предложит установить и настроить пакет Text2speech от Yandex.

Вопросы ecss-restfs	**Ответы**	**Пример**
Хотите ли вы использование функции Text To Speeh (Use TTS service)	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (Configure phone book)	**Yes** (нужно указать)	
Хотите настроить сервер carddav?	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (LDAP)?	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (SSW)?	No (значение по умолчанию)	
Хотите настроить сервис распознавания речи (Configure speech recognition service)	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (POSTGRES)?	Yes (значение по умолчанию)	
POSTGRES: Сервер postgres:	localhost (значение по умолчанию)	
POSTGRES: Порт подключения:	5439 (значение по умолчанию)	
POSTGRES: Логин:	postgres (значение по умолчанию)	

Вопросы ecss-restfs	Ответы	Пример
POSTGRES: Пароль:	postgres1 (значение по умолчанию)	
POSTGRES: База данных:	ecss_storekeeper_db (значение по умолчанию)	
Введите домен для поиска:	test.domain (значение по умолчанию)	
POSTGRES: Количество запрашиваемых контактов:	10000 (значение по умолчанию)	
Хотите включить транслит для телефонной книги (LDAP)?	No (значение по умолчанию)	
Выберите пункты, которые хотите настроить:	**proxy-filter** (необходимо выбрать)	
Включить белый список для api/proxy?		

белый список адресов прописать в файл /etc/ecss/ecss-restfs/api-proxy.conf

```
# ------------------------------------------------------------
# Формат файла фильтра API (api/proxy):
#   - IP: 192.168.0.1
#   - Домен: example.com
#   - Wildcard: *.wildcard.com
#   - Комментарии: строки, начинающиеся с #
# Пример:
#   192.168.0.1
#   example.com
#   # комментарий
#   *.wildcard.com
# ------------------------------------------------------------
```

> **ВАЖНО!**
>
> В случае успешного проксирование (200) кеш хранится 28 дней, поэтому для применения конфигурации фильтра, в случае изменения, - кеш (/var/cache/ecss/restfs-api) стоит удалить и перезапустить сервис ecss-restfs.

после установки пакета ecss-restfs проверьте наличие wav файлов авто-информатора в директории /var/lib/ecss/restfs/system/sounds/ командой:

на обоих хостах

```
ls -la /var/lib/ecss/restfs/system/sounds/
```

наличие wav файлов в указной выше директории говорит о корректности установки пакета ecss-restfs.

Проверить доступность этих файлов извне, для этой проверки выполним команду:

```
wget http://ecss1:9990/system/sounds/ai_you.wav
```

пример

```
wget http://ecss1:9990/system/sounds/ai_you.wav
--2025-09-16 08:27:36--  http://ecss1:9990/system/sounds/ai_you.wav
Распознаётся ecss1 (ecss1)… 10.0.10.40
Подключение к ecss1 (ecss1)|10.0.10.40|:9990... соединение установлено.
HTTP-запрос отправлен. Ожидание ответа… 200 OK
Длина: 11670 (11K) [audio/x-wav]
Сохранение в: «ai_you.wav»

ai_you.wav.1                                  100%
[=============================================================================
===>]  11,40K  --.-KB/s    за 0s

2025-09-16 08:27:36 (81,5 MB/s) - «ai_you.wav» сохранён [11670/11670]
```

проверочные файлы можно удалить

```
rm ai_you.wav
```

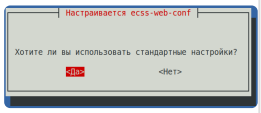
Установка media и web пакетов ecss и их настройка

```
sudo apt install -y ecss-media-server ecss-web-conf
```

Для медиасервера (*ecss-media-server/MSR*) возможно начальное конфигурирование с записью параметров в файл конфигурации,
для этого нужно провести конфигурацию **transport bind-addr,mcc bind-addres**:

Вопросы ecss-media-server	**Ответы для ecss1**	**Пример**
[MSR SIP] Введите bind-ip адрес (Enter)	**10.0.20.40** (необходимо ввести)	
[MSR Control-Channel] Введите bind-ip-адрес	**10.0.20.40** (необходимо ввести)	

Web-конфигуратор позволяет сделать управление системой более наглядным и комфортным. Установка web-конфигуратора не является обязательной, но рекомендуется.
Также при установке пакета **ecss-web-conf** автоматически устанавливается пакет **ecss-subsriber-portal-ui**. Приложение "Портал абонента" системы ECSS-10 позволяет абонентам системы самостоятельно управлять услугами, просматривать информацию по совершенным вызовам, активным конференциям, а также настраивать собственные IVR-скрипты для входящих вызовов. Описание работы веб-конфигуратора приведено в разделе "Портал абонента".

Вопросы ecss-web-conf	**Ответы для ecss1/ecss2**	**Пример**
Хотите ли вы использовать стандартные настройки?	Yes (значение по умолчанию)	

После формирования конфигураций по умолчанию переходим в директорию, в которой находятся файлы конфигурации и производим их проверку:

```
cd /etc/ecss/ecss-media-server/
```

```
cat /etc/ecss/ecss-media-server/config.xml
```

Конфигурация для msr: файл - config.xml, в директории conf.d файл default.xml.
По своей сути default.xml — это дополнение config.xml, которое определяет секцию аккаунтов. Это сделано для того, чтобы после обновлений пакета данная конфигурация оставалась неизменной. Вид config.xml представлен здесь: файл конфигурации.

```
<?xml version="1.0" encoding="utf-8"?>
<config date="02:30:33 07.12.2023">
  <general log-level="3" log-rotate="yes" max-calls="8192" max-in-group="512" load-sensor="medi
a" load-delta="10" calls-delta="100" spool-dir-size="100M" log-name="msr.log" log-path="/var/
log/ecss/media-server" use-srtp="disabled" enable-ice-transport="no" ice-update="no"
 aggressive-ice="yes" stun-server="" suspicious-mode="no"/>
  <transport bind-addr="10.0.20.40" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca" user="webrtc@live.com" password="muazkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60" rtcp-timeout="0" rtp-
timeout="350" udp-src-check="no" cn-multiplier="3" port-start="12000" port-range="2048" tias-
in-sdp="no" thread-cnt="2" vid-enc-threads="2" vid-dec-threads="2" video-conf-layout="evenly"
 keyframe-interval="1000" vid-decode-delay="100" silence-threshold="-30" dtmf-flash-disable="no
" video-dscp="0" other-dscp="0" dummy-video-src="/usr/share/ecss-media-server/video/
```

```
dummy_video.yuv" video-enc-width="640" video-enc-height="360" finalsilence="1000" rtcp-stat-
dump="yes" silent-codec-switch="yes"/>
  <codec pcma="1" pcmu="2" ilbc="0" gsm="0" g722="3" g726="0" g729="0" speex="0" l16="0" g7221="
0" opus="0" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
  <accounts>
    <!-- <dynamic msr_name="msr.name"
            realm="sip:127.0.0.1:5000"
            dtmf_mode="rfc+inband+info"
            auth_name="user"
            auth_password="password" /> -->
  </accounts>
  <pbyte>
    <mcc bind-addr="10.0.20.40" port="5700"/>
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

По умолчанию после инсталляции ECSS-10 **активизирует только** кодеки — **pcma, pcmu, g722,** h264, h263-1998, t38. Остальные имеют статус =0 = выкл. **Если требуется** активировать остальные кодеки, в редакторе nano измените приоритет с 0 на цифру приоритета (уровень приоритета не должен повторятся для сервиса audio/video/fax)
sudo nano /etc/ecss/ecss-media-server/config.xml

чтобы активизировать **все** кодеки
строчку -
<codec pcma="1" pcmu="2" ilbc="**0**" gsm="**0**" g722="3" g726="**0**" g729="**0**" speex="**0**" l16="**0**" g7221="**0**" opus="**0**" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
привести к виду -
<codec pcma="1" pcmu="2" ilbc="**6**" gsm="**4**" g722="3" g726="**11**" g729="**5**" speex="**7**" l16="**8**" g7221="**9**" opus="**10**" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>

Недопустимо указывать у разных кодеков одинаковый приоритет, для примера g722="5" g729="5". В этом случае media-server не сможет работать корректно.
За исключением "0" - выключено.

Рассмотрим секцию аккаунтов (файл default.xml):

```
cat /etc/ecss/ecss-media-server/conf.d/default.xml
```

После инсталляции пакетов в фале будет указан ip-add 127.0.0.1.

```
<?xml version="1.0"?>
<config>
      <accounts>
            <dynamic msr_name="msr.ecss1" realm="sip:127.0.0.1:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>
      </accounts>
</config>
```

Требуется указать в конфигурации, реальный IP-add.

Необходимо привести содержимое файла к виду указанному снизу (в примере ecss1 имеет IP-add 10.0.20.40):

на обоих хостах

```
sudo nano /etc/ecss/ecss-media-server/conf.d/default.xml
```

Настройка msr для ecss1 (/etc/ecss/ecss-media-server/conf.d/default.xml)

Проверить на корректность данных.

```
<?xml version="1.0"?>
<config>
  <accounts>
    <dynamic msr_name="msr.ecss1" realm="sip:10.0.20.40:5000" dtmf_mode="rfc+inband+info"
 auth_name="user" auth_password="password">
      <via dynamic-ifaces="no">
        <iface name="net.20" ip="10.0.20.40"/>
      </via>
    </dynamic>
  </accounts>
</config>
```

Указываем текущие настройки, согласно которым происходит регистрация msr на core.

Основными параметрами здесь являются: msr_name и realm:

- msr_name — параметр, определяющий название msr. (рекомендуется задавать msr. и к какому хосту он принадлежит, например msr.ecss1);
- realm — определяет адрес для регистрации на ядре. Точка входа по умолчанию — порт 5000.

После изменения конфигурации файлов **/etc/ecss/ecss-media-server/config.xml** и **/etc/ecss/ecss-media-server/conf.d/default.xml** необходимо выполнить перезапуск сервиса **ecss-media-server**, чтобы изменения вступили в силу, следующей командой:

```
sudo systemctl restart ecss-media-server
```

Описание настройки остальных сервисов приведено в разделе "Инсталляция дополнительных необязательных пакетов".

Дальнейшая настройка

snpmd

Настройка SNMP описана в разделе "Управление и мониторинг по протоколу SNMP".

Настройка безопасности. SSH

Выполните настройку сервера SSH:

```
sudo nano /etc/ssh/sshd_config
```

Укажите в конфигурационном файле порт и адрес, по которому можно обратиться к серверу:

Настройка ssh для ecss1 (/etc/ssh/sshd_config)

```
# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Port 2000
#AddressFamily any
ListenAddress 10.0.10.40
#ListenAddress ::

<...>
```

Перезапустите ssh:

```
sudo systemctl restart ssh.service
```

Перезагрузка хоста

```
sudo reboot
```

Добавление услуг

Установка всех услуг указных в лицензии:

выполняется в CoCon

```
/cluster/storage/ds1/ss/install ds1@ecss1 ss_*
```

Результат:

```
/cluster/storage/ds1/ss/install ds1@ecss1 ss_*
Successfully installed: /var/lib/ecss/ss/ss_dnd.xml
Successfully installed: /var/lib/ecss/ss/ss_clir.xml
...
Successfully installed: /var/lib/ecss/ss/ss_chunt.xml
```

11.6 Первоначальная конфигурация системы с помощью web-конфигуратора

пример первоначальной конфигурации через Web интерефейс

11.6.1 Исходные данные

- Установка системы завершена;
- Система готова к дальнейшей настройке;
- Интерфейсы запущены.

Рекомендуется использовать последние доступные версии браузеров. Рекомендуемые к использованию браузеры: Яндекс, Chrome.

Для начала конфигурации системы необходимо зайти в web-конфигуратор по адресу http://<Ipadd SSW>.

Для определения и регистрации в системе планируются:

- Абоненты с номерами 42001, 42002 , 42003, 42004 ,42005, 42006, 42007, 42008, 42009, 42010;
- Транк в сторону шлюза.

11.6.2 Подготовка к работе

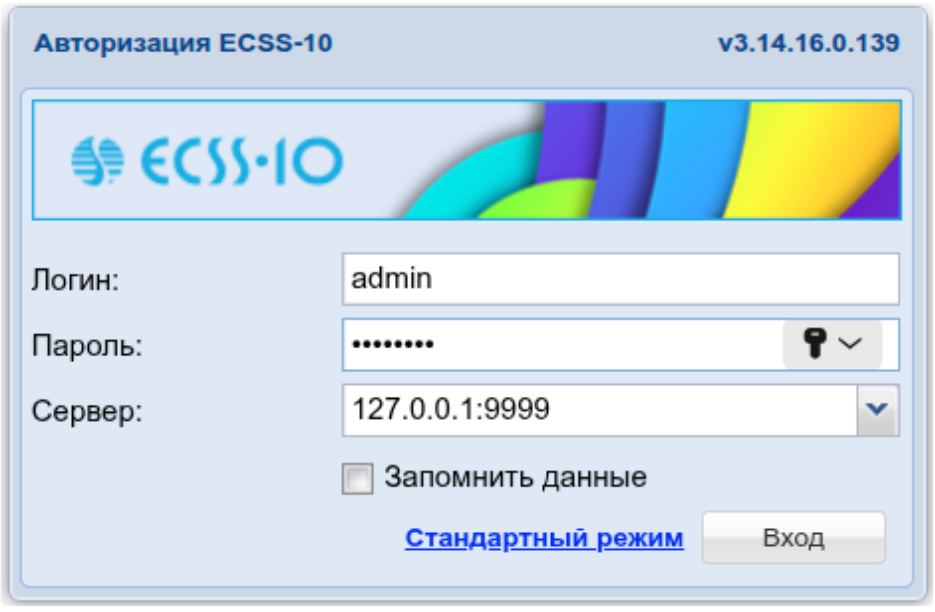


Рисунок 3 — Вход в web-конфигуратор (окно авторизации)

В окне авторизации необходимо ввести значения, определенные при установке web-конфигуратора.

Стандартные значения для авторизации:
Логин: admin
Пароль: password

После входа в систему будет видна основная рабочая область с иконками приложений, а также строка состояния с доступными опциями, в частности:

- 1 — выход из системы;
- 2 — выбор домена;
- 3 — выбор языка.

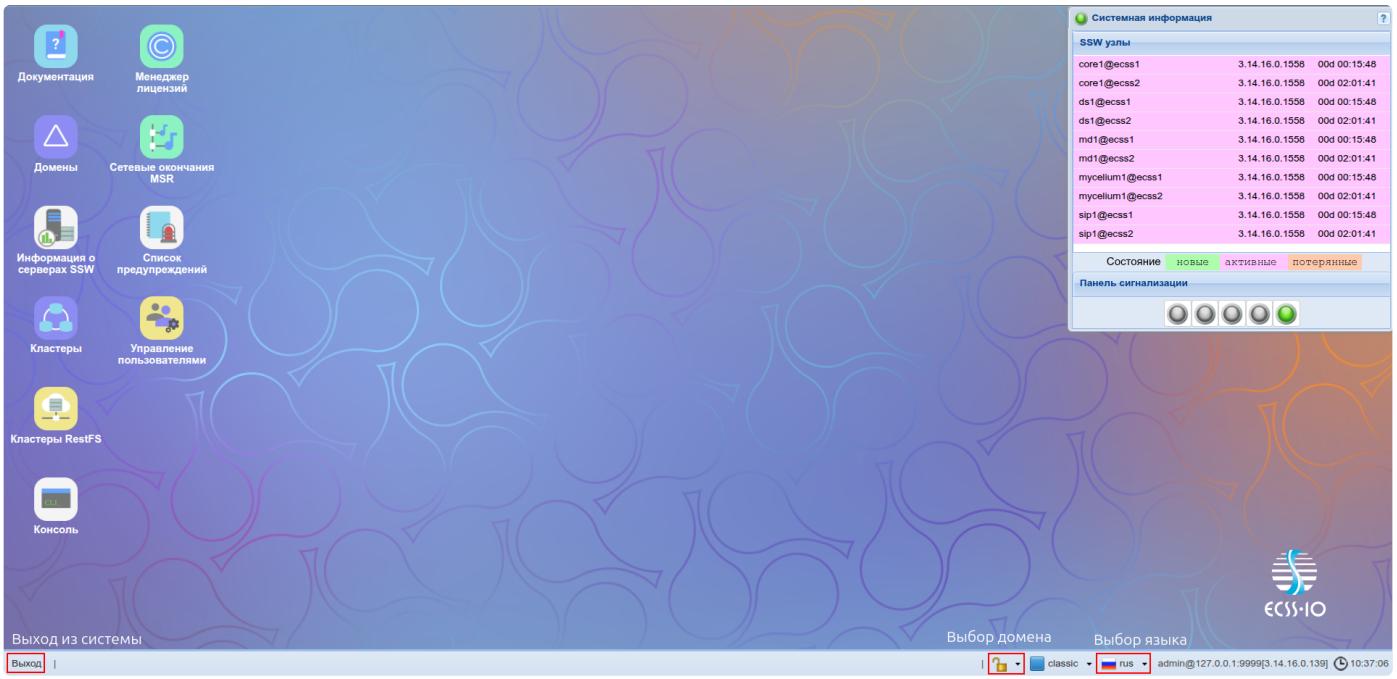


Рисунок 4 — Вид рабочей области web-конфигуратора

11.6.3 Создание учетной записи оператора

После авторизации для повышения безопасности при эксплуатации программного коммутатора рекомендуется создание учетных записей для операторов, а также сменить пароль для пользователя **admin**.

Для создания новой учетной записи оператора используется приложение "Управление пользователями" ("User manager"):

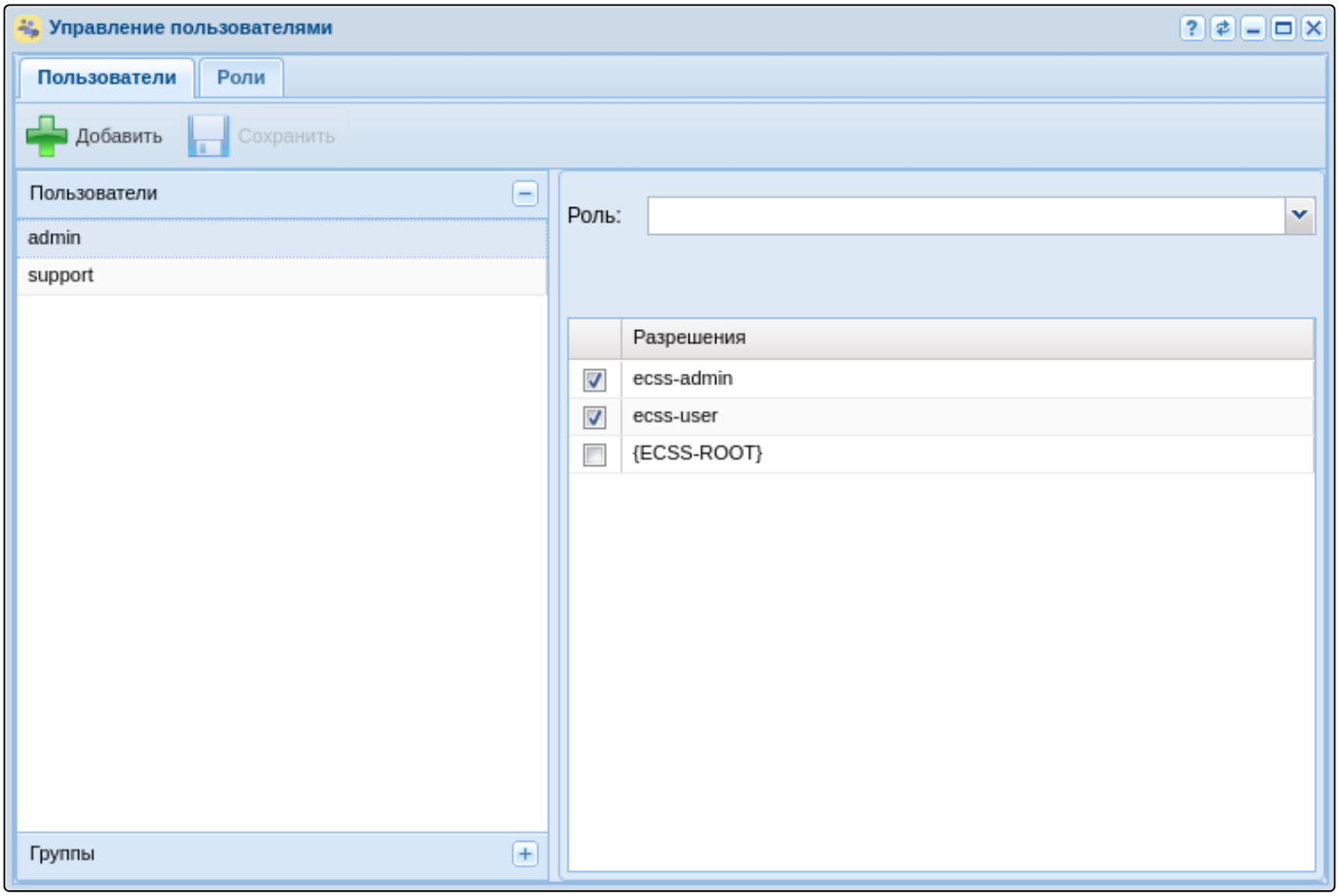


Рисунок 5 – Вид приложения "*Управление пользователями*"

Нажмите кнопку "*Добавить*"  . В открывшемся окне требуется определить новую учетную запись, для этого:

1. В поле "*Имя*" введите логин учетной записи, например "test";
2. В поле "*Пароль*" и "*Подтверждение*" введите пароль для пользователя, например "testpassword";
3. Определите уровень прав доступа для пользователя через выбор текущих разрешений или с помощью ролей, например **ecss-user**. Описание можно посмотреть в разделе "Управление пользователями".

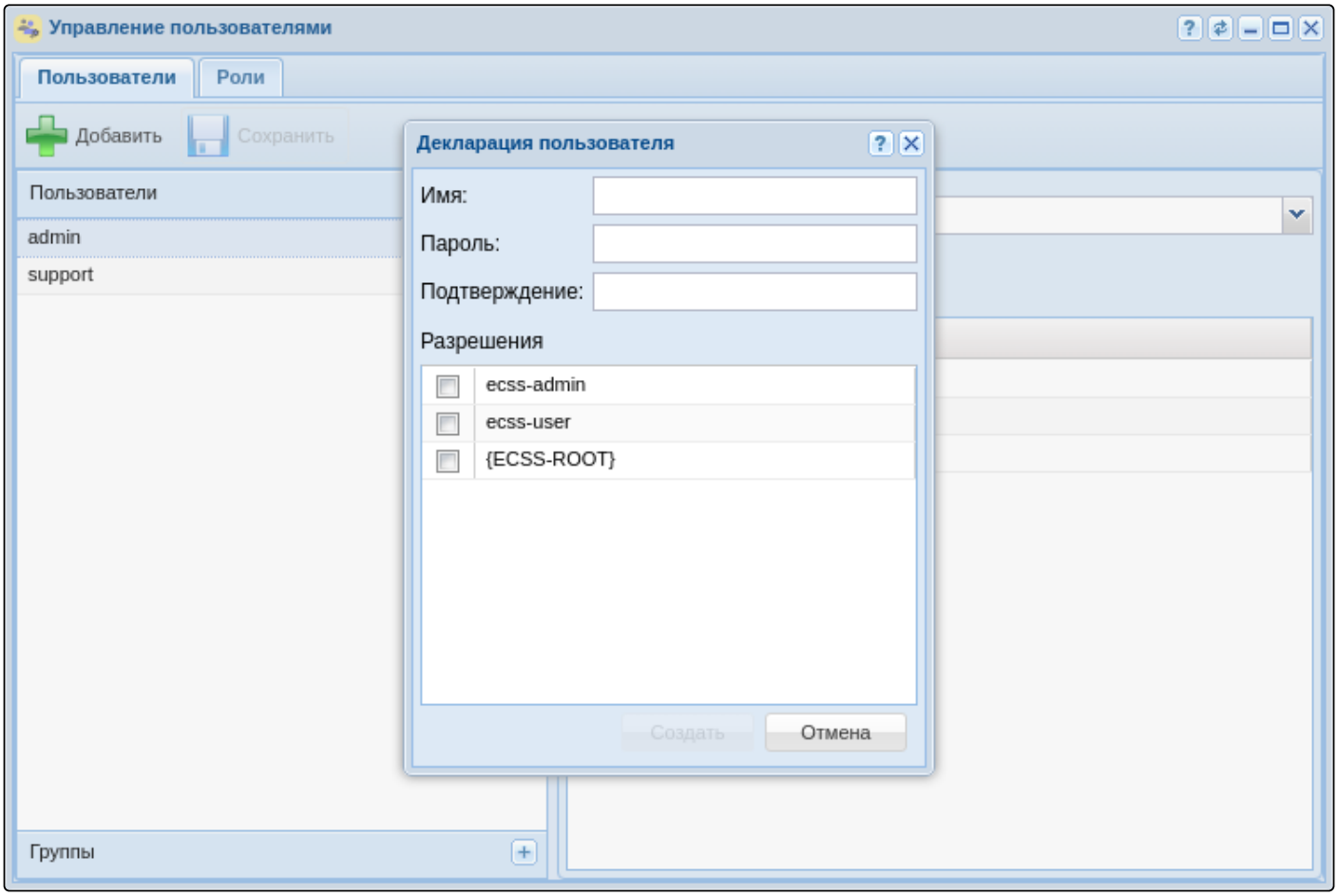


Рисунок 6 – Диалоговое окно создания учетной записи оператора

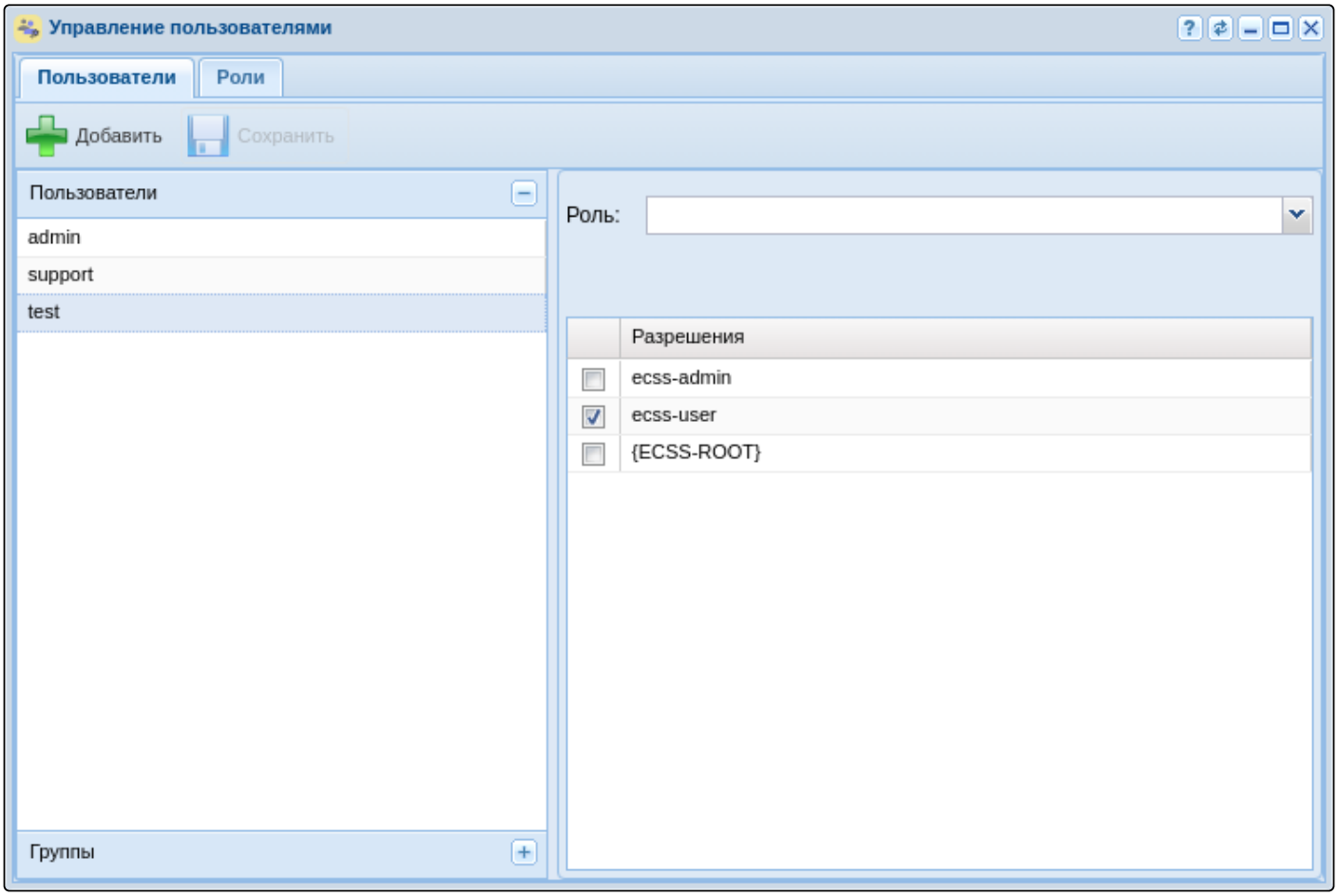


Рисунок 7 — Вид приложения с созданной учетной записью оператора

Для смены пароля рядом с названием пользователя нажмите на кнопку редактировать. В появившемся диалоговом окне введите:

1. Старый пароль (для пользователя "**admin**" по умолчанию паролем является **"password"**);
2. Новый пароль;
3. Подтвердите новый пароль.

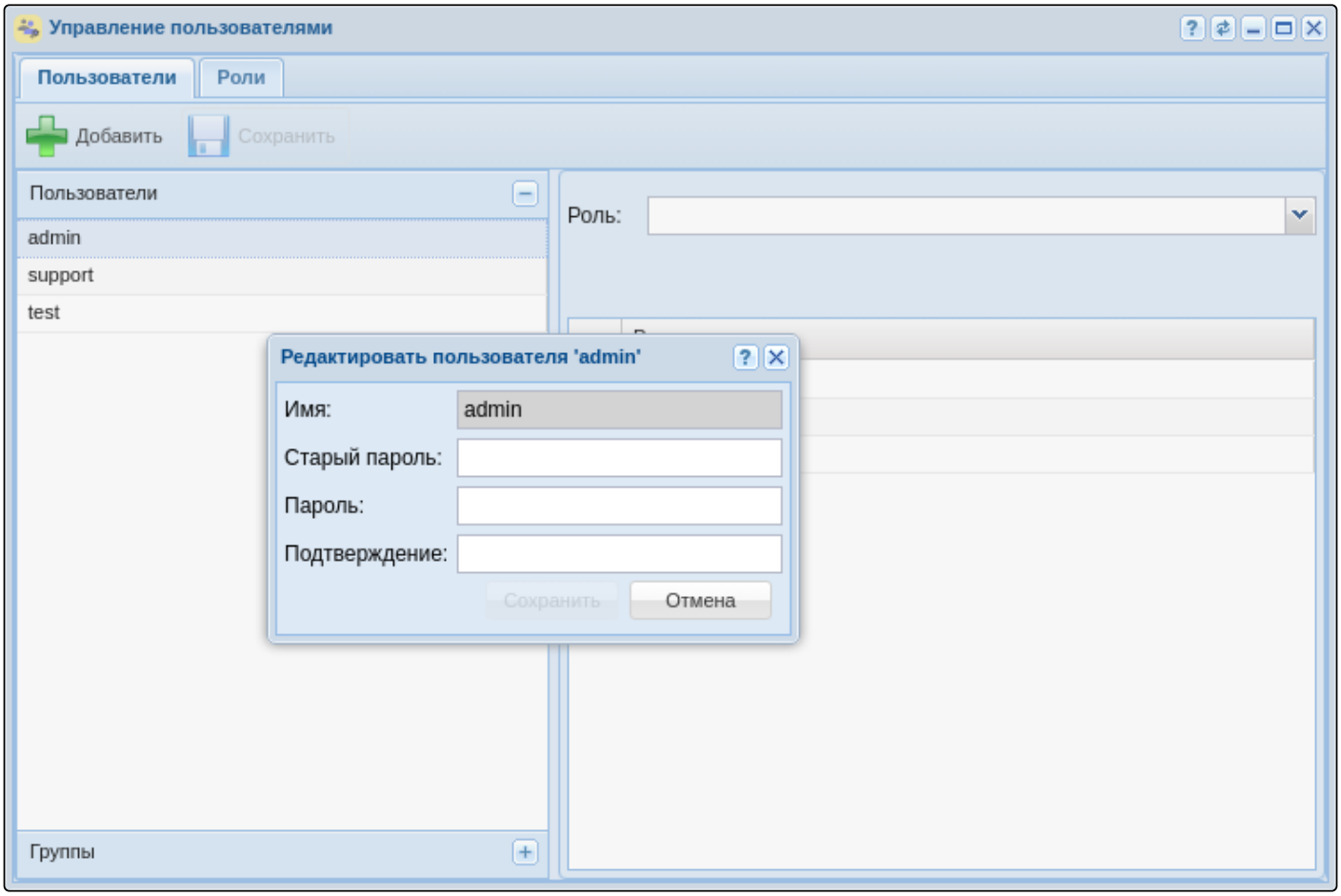
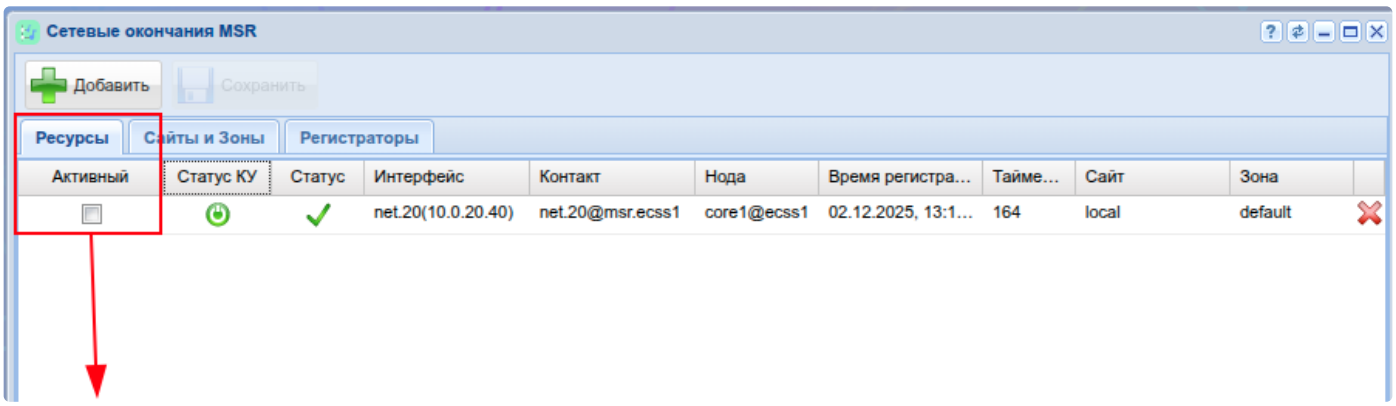


Рисунок 8 — Диалоговое окно редактирования учетной записи

11.6.4 Добавление медиа-ресурсов.

Для того чтобы начать работу с медиа-сервером сразу после его регистрации, необходимо активировать сетевые интерфейсы MSR. Приложение web-конфигуратора "Сетевые окончания MSR" ("MSR registrars") служит для настройки медиа-ресурсов.



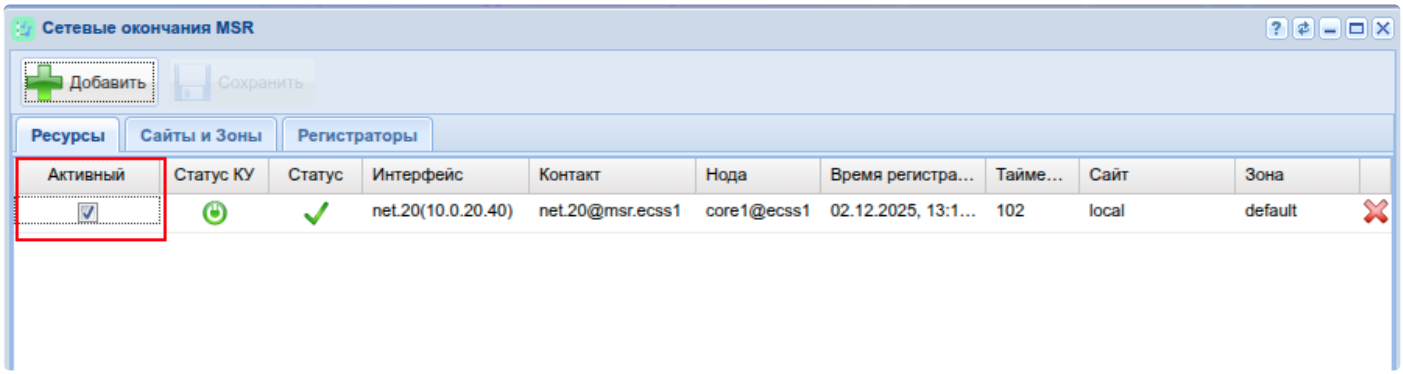
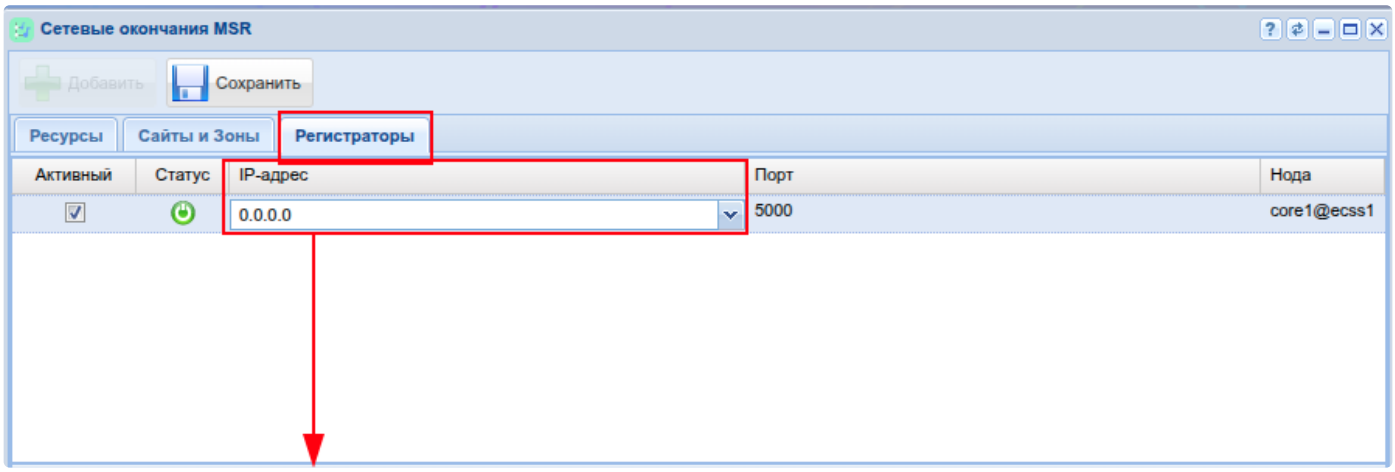
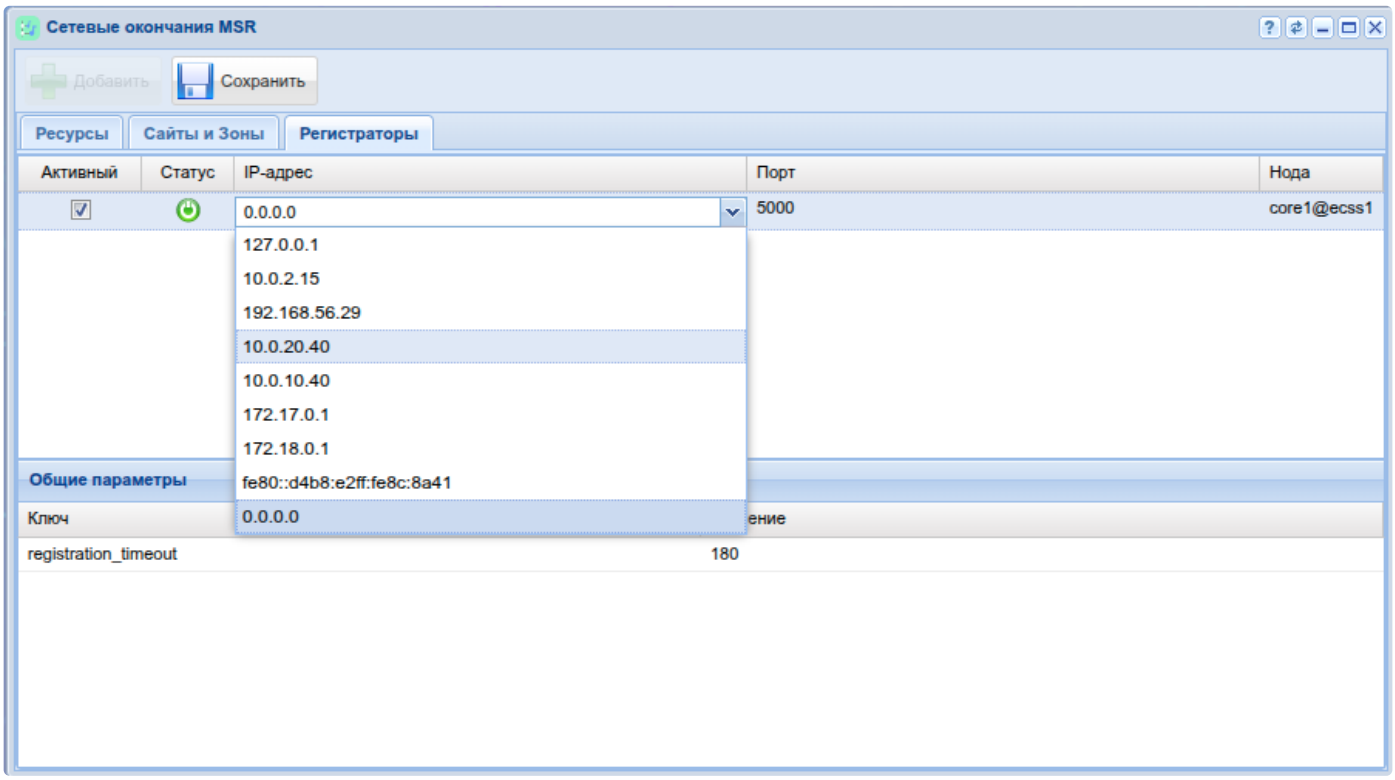


Рисунок 9 – Сетевые окончания MSR

Установить ipадрес для приема соединений от медиасервера, изменив значение 0.0.0.0 на реальные адреса MSR:





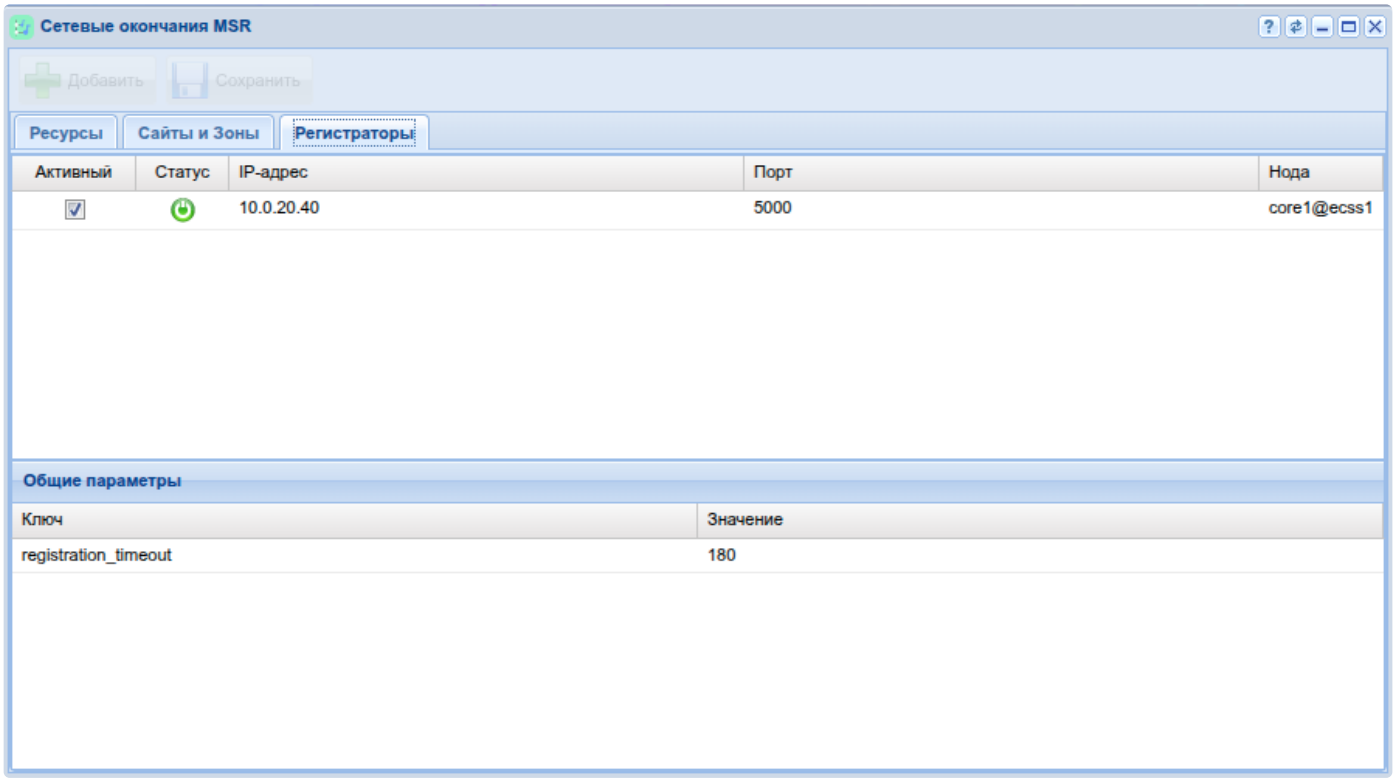


Рисунок 10 — Установка IP-add регистратора MSR

11.6.5 Создание IP-set (sip-транспорта) и присвоение его к домену

IP-set — совокупность одного или нескольких IP-адресов из одной подсети и списка слушающих UDP/TCP-портов, которые будут открыты SIP-адаптером на данных адресах.

Для настройки интерфейса необходимо открыть приложение "Кластеры ("Clusters").

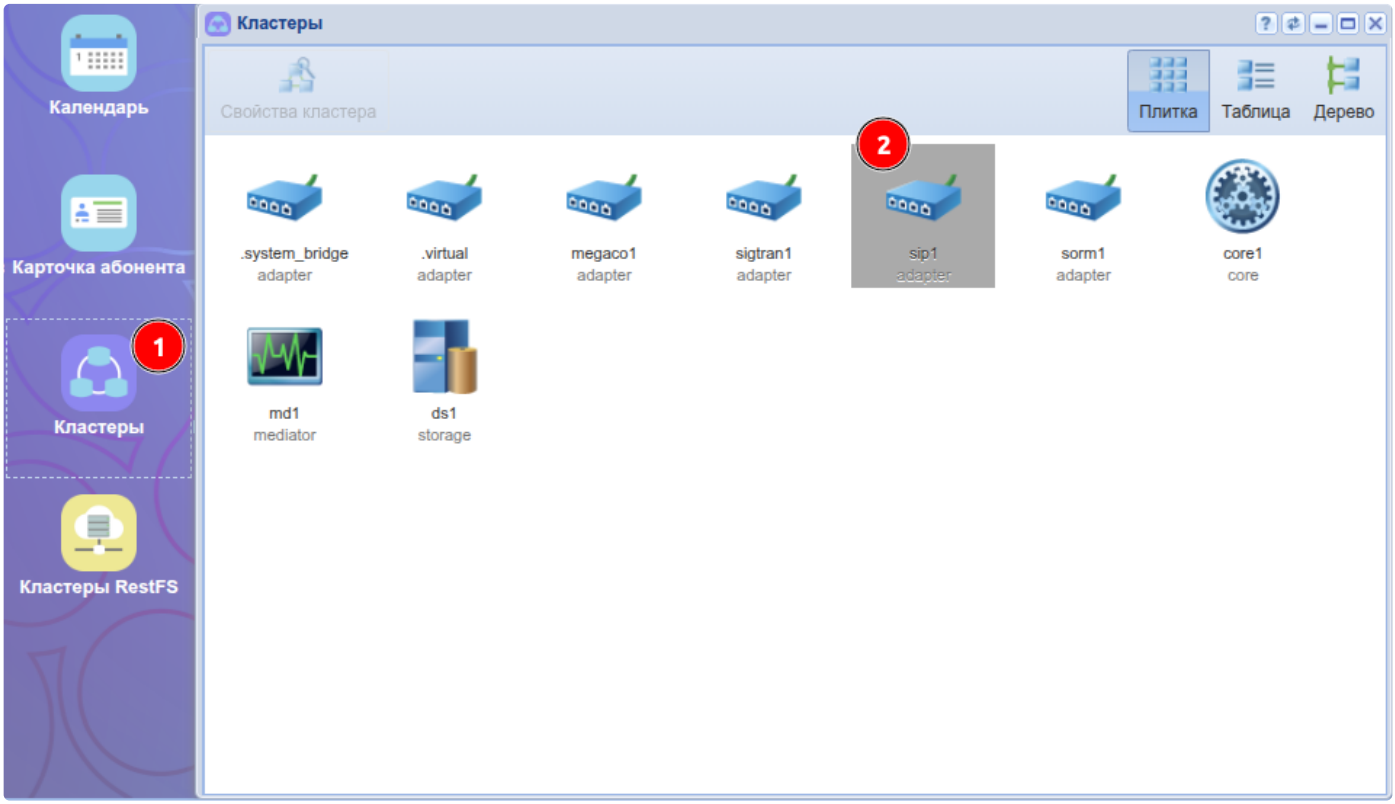


Рисунок 11 — Вид приложения "Кластеры"

Для создания новой группы IP-адресов (IP-set) выделите кластер SIP-адаптера "sip1" и нажмите на кнопку *"Свойства кластера"* (или нажмите на иконку кластера двойным щелчком левой кнопки мыши).

В появившемся диалоговом окне перейдите на вкладку *"Транспорт"*. Далее нажмите кнопку *"Добавить"*. Появится новая группа, для редактирования полей двойным щелчком кнопки мыши нажмите на нужное:

1. Переименуйте группу адресов (IP-set), например "test_set";

2. Задайте порт, по которому будет происходить обращение к домену, например 5060;

3. Разверните вновь созданную группу нажатием на треугольник слева от названия группы;

4. Определите адрес для ноды SIP-адаптера, согласно примеру конфигурации. Для примера 10.0.20.40:

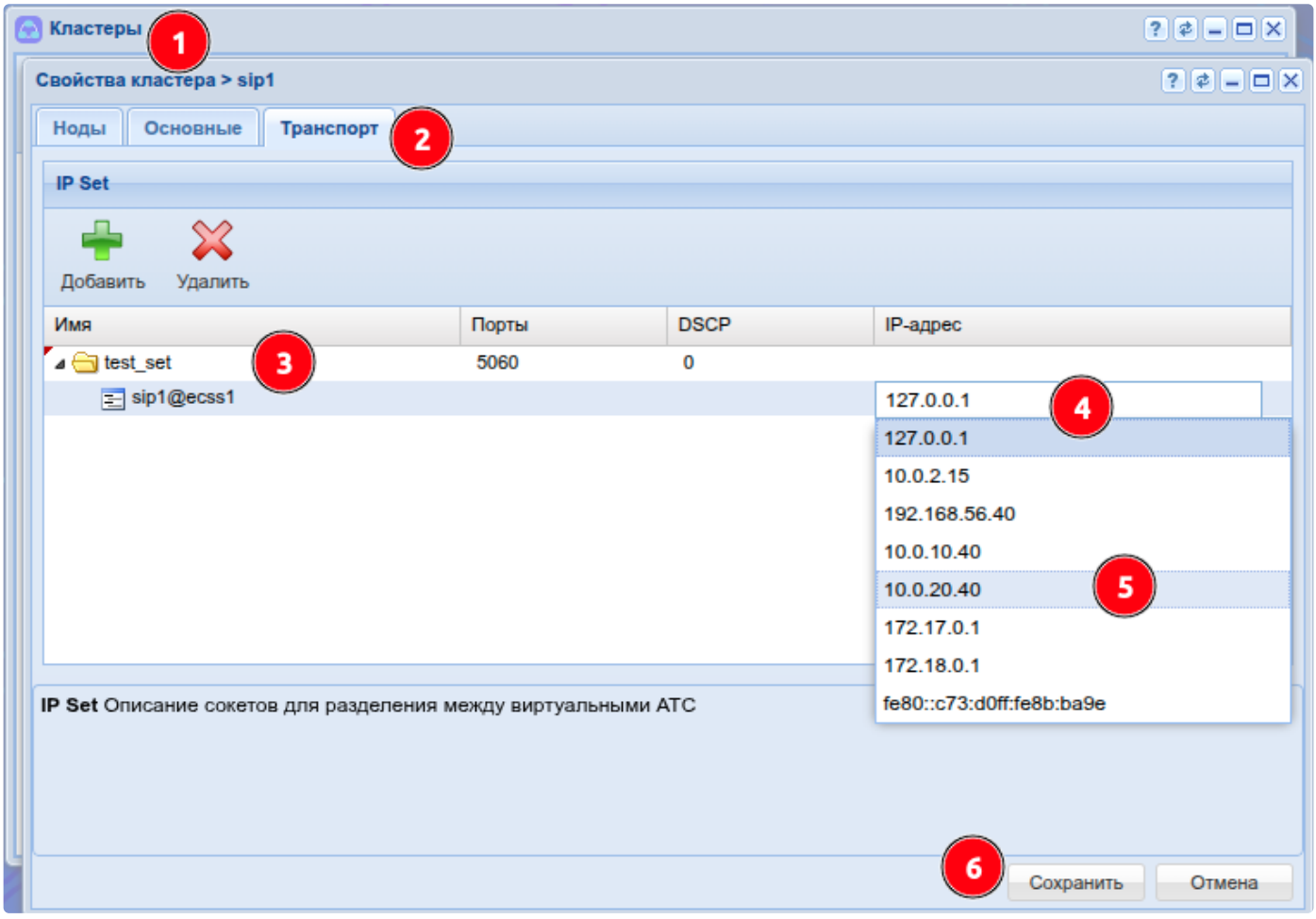


Рисунок 12 — Вид приложения "Кластеры/Транспорт"

Нажмите кнопку *"Сохранить"* для применения настроек.

11.6.6 Создание домена

Домен - Виртуальная PBX внутри нашего SSW. Таких виртуальных PBX может быть создано необходимое для проекта кол-во.

Для создания домена необходимо войти в приложение *"Домены" ("Domain")*. В открывшемся окне требуется создать домен, для этого:

1. Нажмите кнопку "*Добавить домен*":

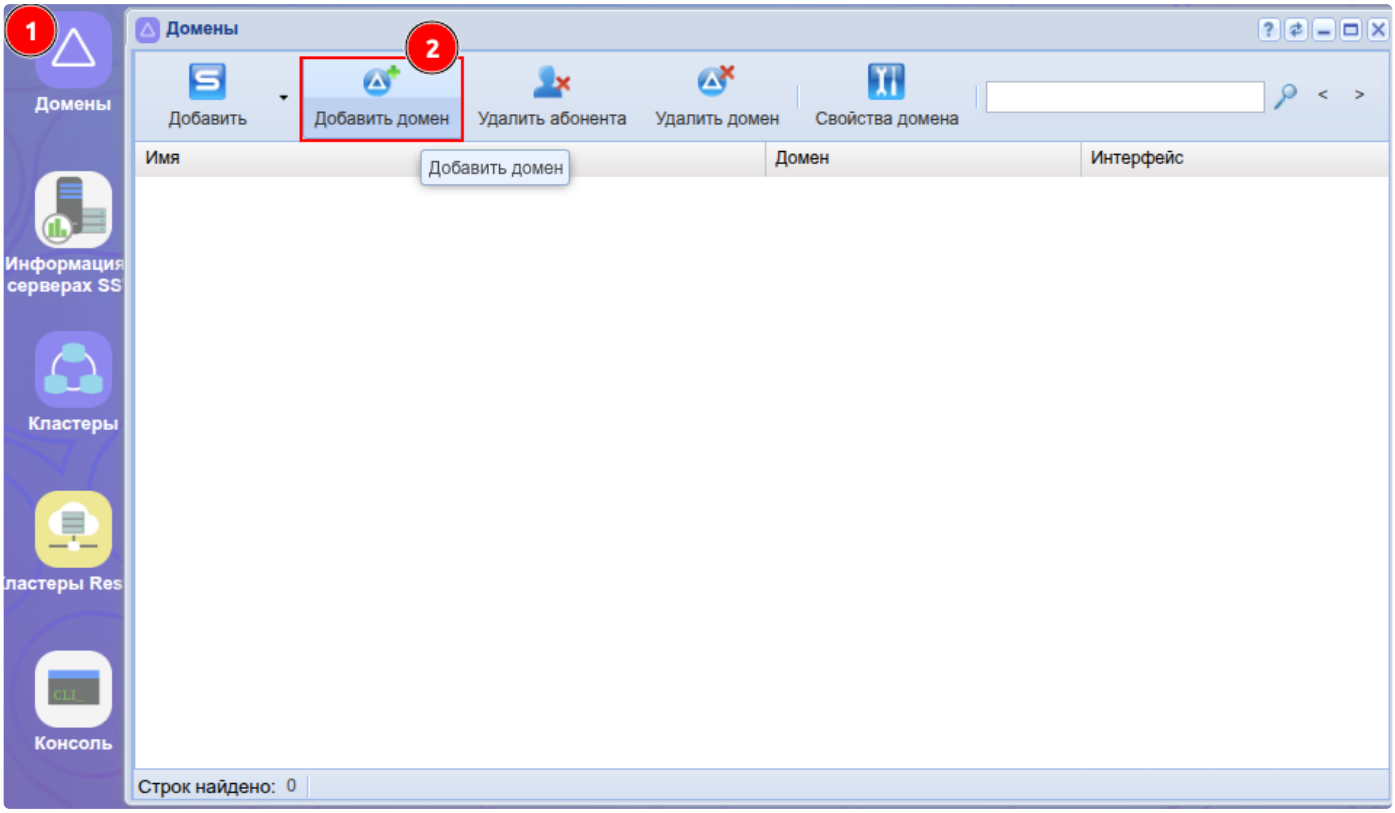


Рисунок 13 — Добавление домена в систему

2. В появившемся диалоговом окне доступны следующие настройки:

- *Имя (Name)* — индивидуальное имя виртуальной АТС;
- *Профиль услуг (SS profile)* — системный профиль дополнительных услуг. Этот профиль будет скопирован с тем же именем во вновь создаваемый домен и для всех услуг из этого профиля будет автоматически разрешён доступ через access-list;
- *IVR профиль (IVR profile)* — профиль IVR, заданный в приложении "Редактор IVR ограничений".

Введите название домена, например "test_domain";

3. Нажмите кнопку "*Ок*":

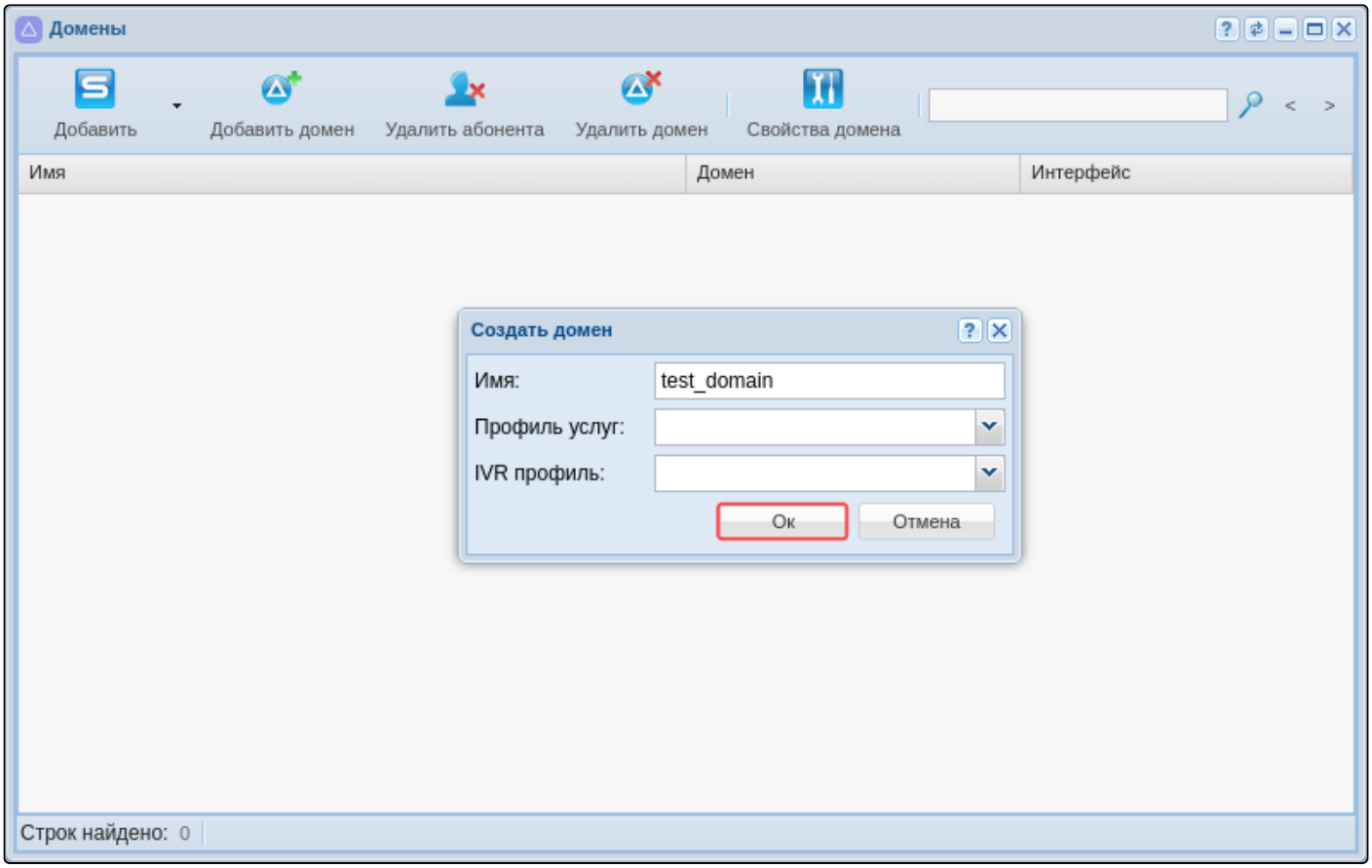


Рисунок 14 – Определение настроек нового домена

4. Нажмите кнопку *"Обновить"* в приложении.

Созданный домен будет отображаться в текущей конфигурации:

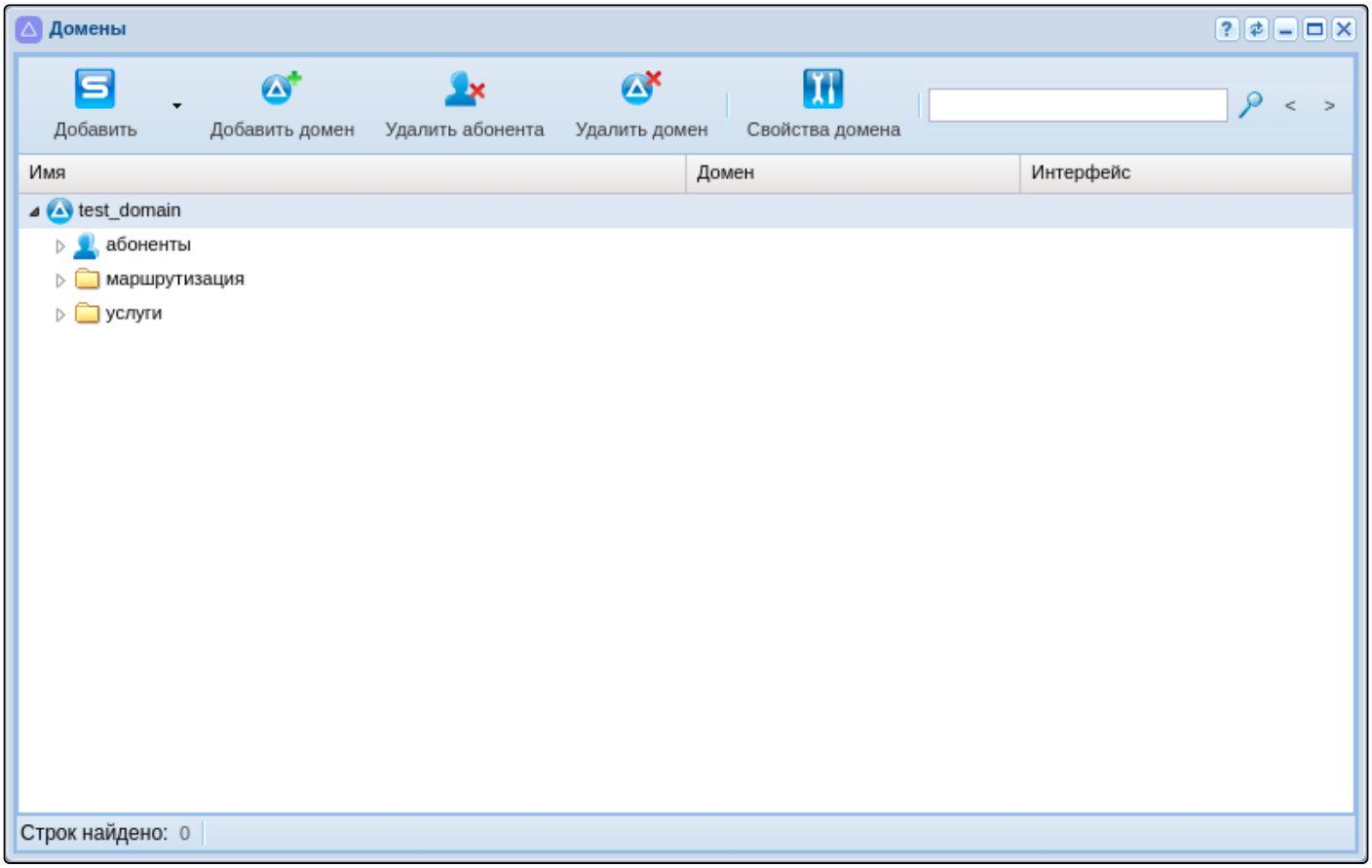


Рисунок 15 — Отображение созданного домена

Для редактирования текущего домена его необходимо выбрать в системе. Для перехода в домен воспользуйтесь опцией выбора домена (см. пункт 2 на рисунке "Вид рабочей области web-конфигуратора").

После выбора домена, согласно текущей конфигурации системы, станут доступны все приложения:



Рисунок 16 — Отображение приложений в текущей конфигурации системы

Для того чтобы **привязать группу адресов к домену**, необходимо вернуться в приложение *"Домены" ("Domain")*, выделить домен и перейти в настройки нажатием на кнопку *"Свойства домена"* или двойным нажатием левой кнопки мыши на домен.

В настройках списка откройте ветку "SIP", в ней "SIP транспорт", далее выберите в поле IP set созданную группу адресов. Нажмите кнопку *"Сохранить"* для применения настроек.

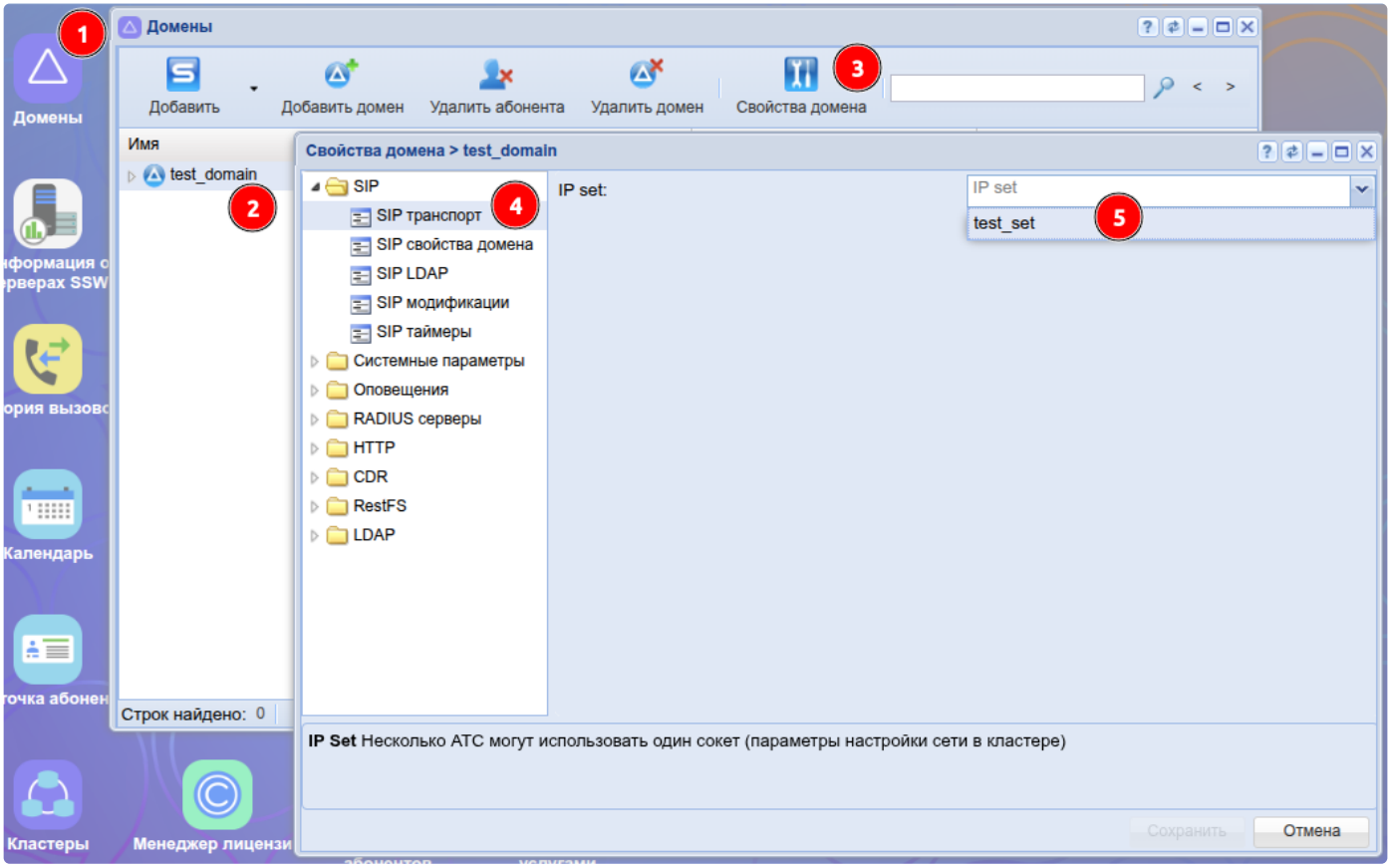


Рисунок 17 — Окно настройки для SIP транспорта

Выбрать требуемые услуги для созданного домена, которые будут использоваться (для примера для абонентского домена транковые услуги не востребованы , и наоборот для транзитного домена не нужно указывать услуги для абонента).

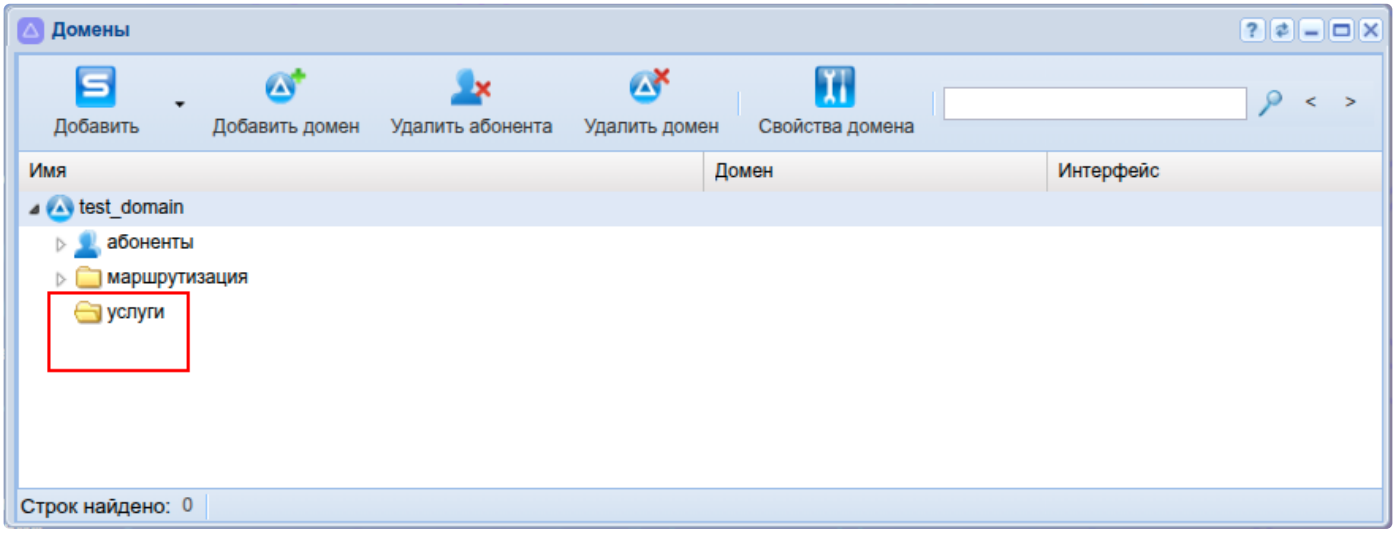


Рисунок 18 — После создания нового домена, услуги отсутствуют.

1. Выбрать приложение "Управление услугами"
2. Выбрать колонку "Установленные услуги" (услуги указанные в лицензии для SSW)

3. Выбрать необходимые для указанного домена услуги
4. Сохранить конфигурацию

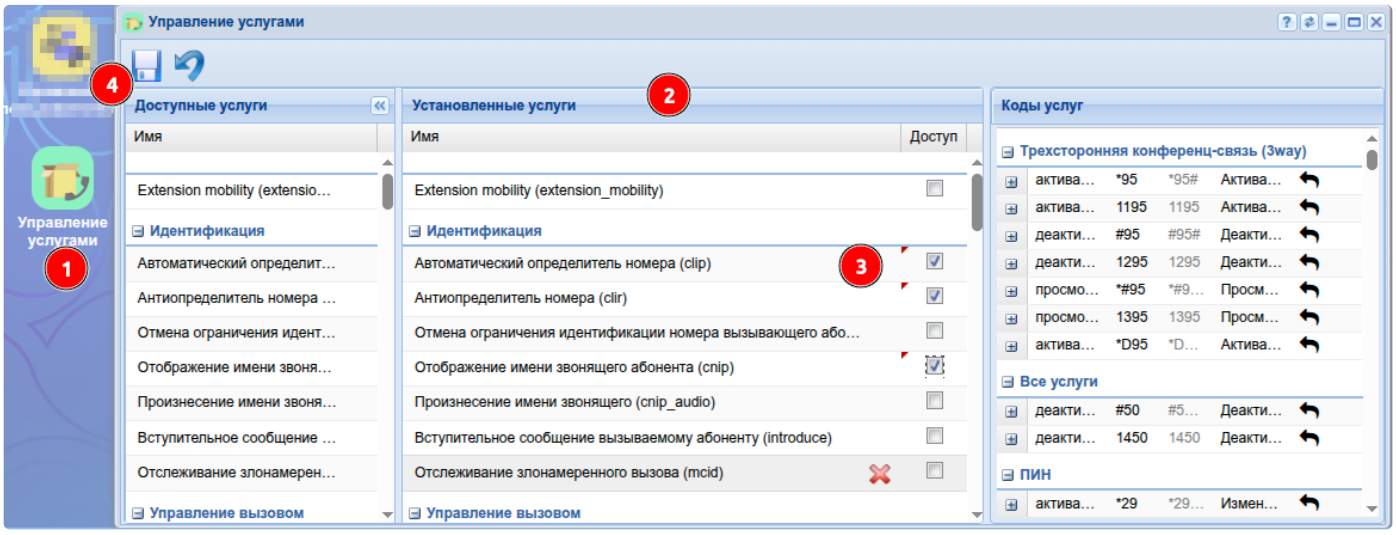


Рисунок 19 — Выбор услуг для абонентов / транков нового домена.

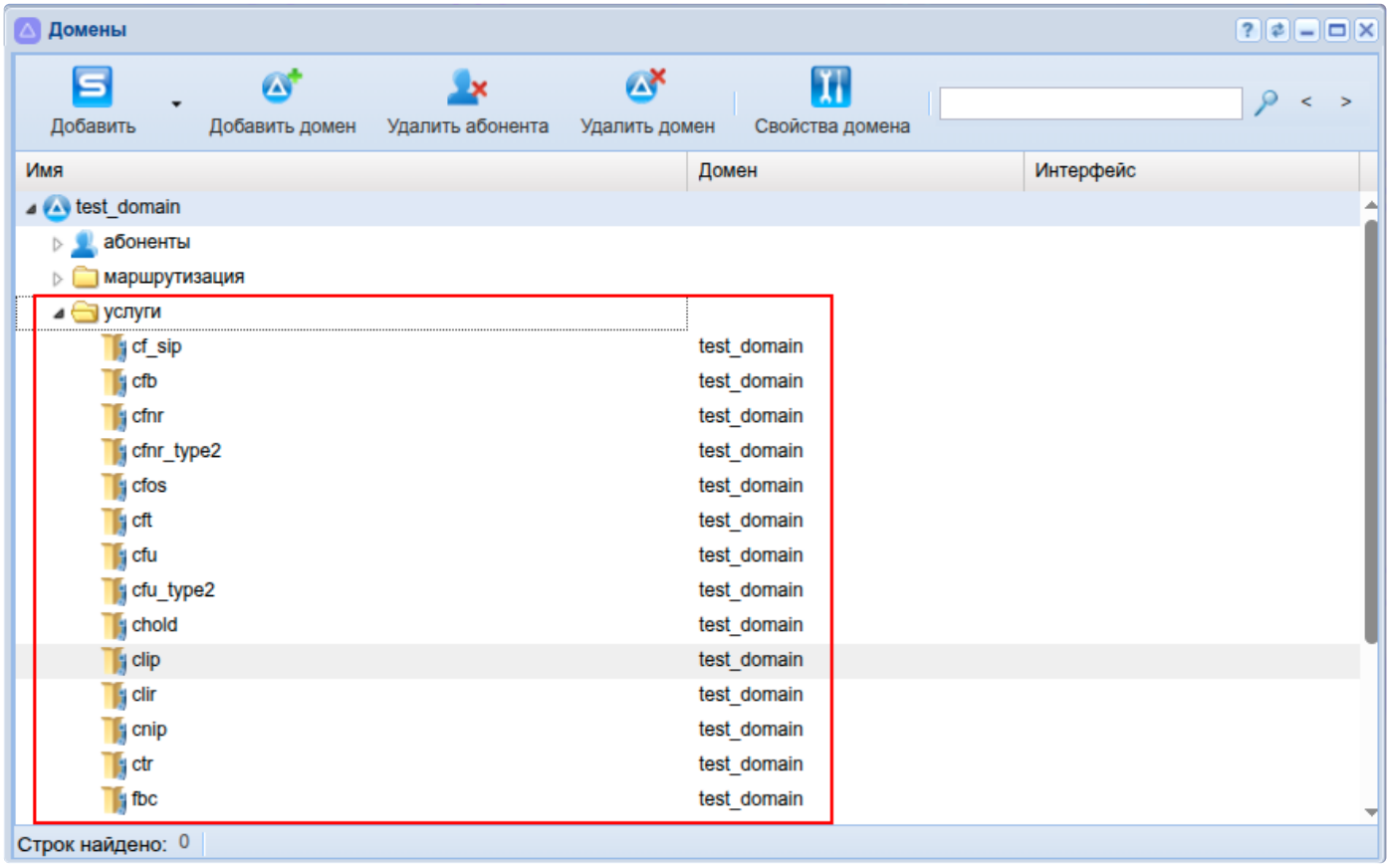


Рисунок 20 — Список услуг теперь доступных для абонентов нового домена.

11.6.7 Создание абонентов

Для создания и редактирования параметров абонентов в системе используется приложение "Карточка абонента" ("Subscriber card").

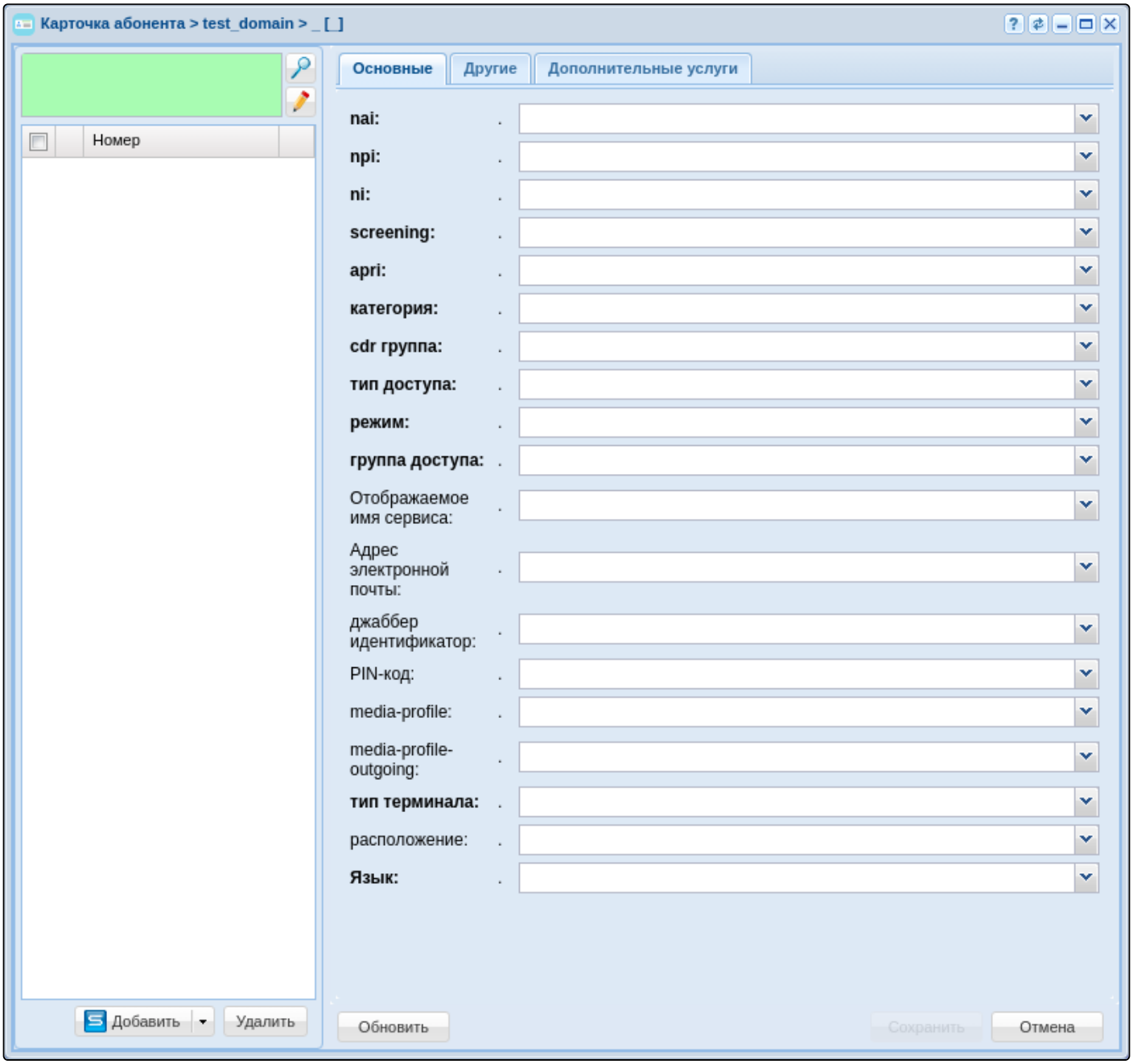


Рисунок 21 — Вид приложения "Карточка абонента"

В конфигурации возможно создание SIP-абонентов и виртуальных абонентов.

Для пользователей, имеющих физическое окончание, используется функционал SIP-абонента, в то время как виртуальный абонент используется тогда, когда нужна функциональность без физических окончаний. Например, номер для обращения к ivr-скрипту.

Для создания новых абонентов нужно нажать на кнопку *"Добавить"*.

В появившемся диалоговом окне укажите следующие параметры:

- *Имя интерфейса* — номер или группа номеров, который присваивается абоненту, строка, например {42001-42010} или abcd;
- *Контекст* — контекст маршрутизации, выбираем тот, который создали, например "test_name", или по умолчанию - "default_routing";
- Группа интерфейсов — имя группы объединяющей нескольких абонентов, для примера "sip";
- *Алиас как пользователь* — настройка, которая связывает номер сущности, алиас и пользователя одним и тем же названием, в примере настройка активирована;

- *Авторизация* — порядок проверки подлинности прав пользователя на доступ к данным, в примере используется"register";
- *Логин* — использую настройку номер как логин для простоты;
- *Пароль* — можно задать или использовать сгенерированный системой.

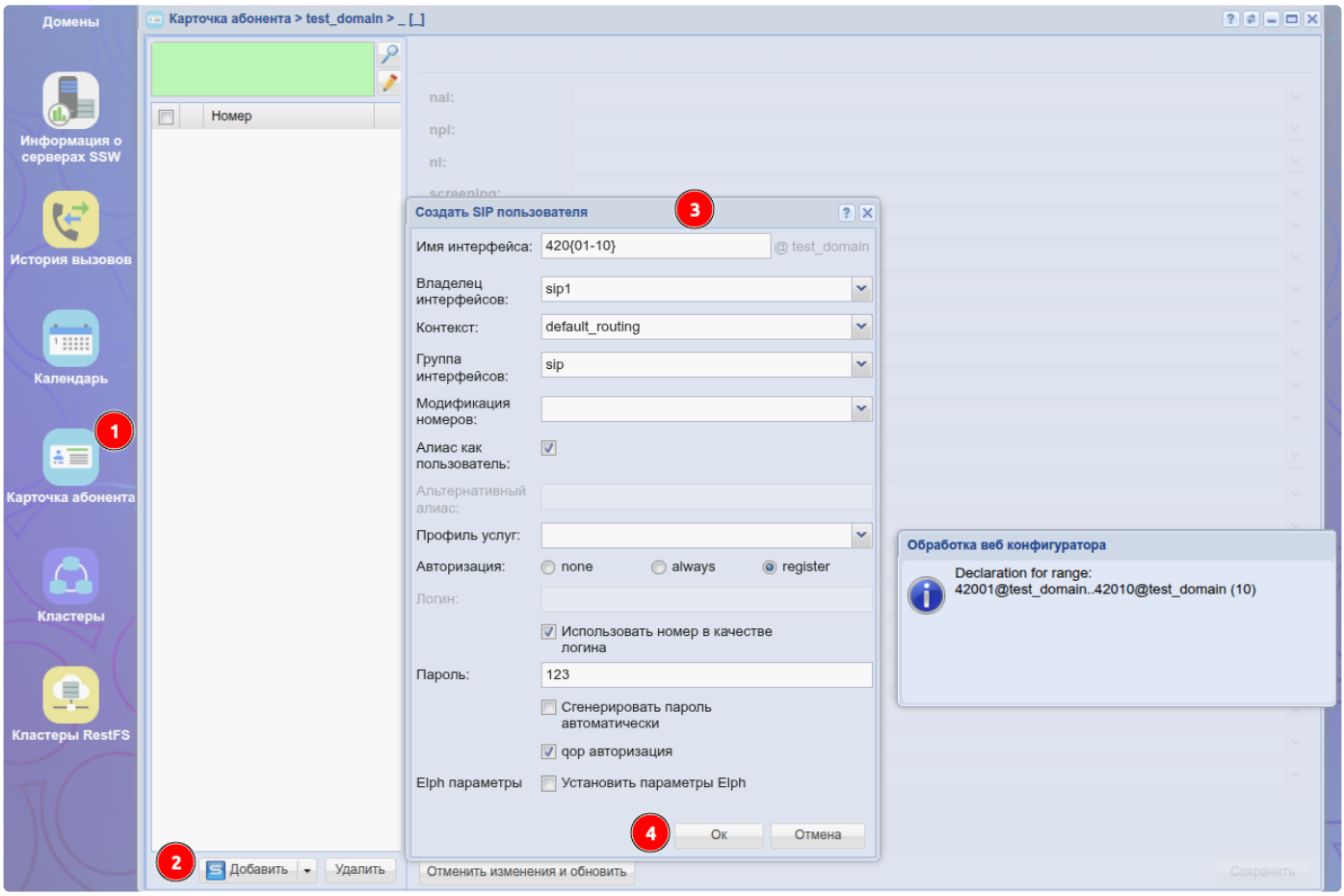


Рисунок 22 — Пример создания 10 абонентов в домене "test_domain"

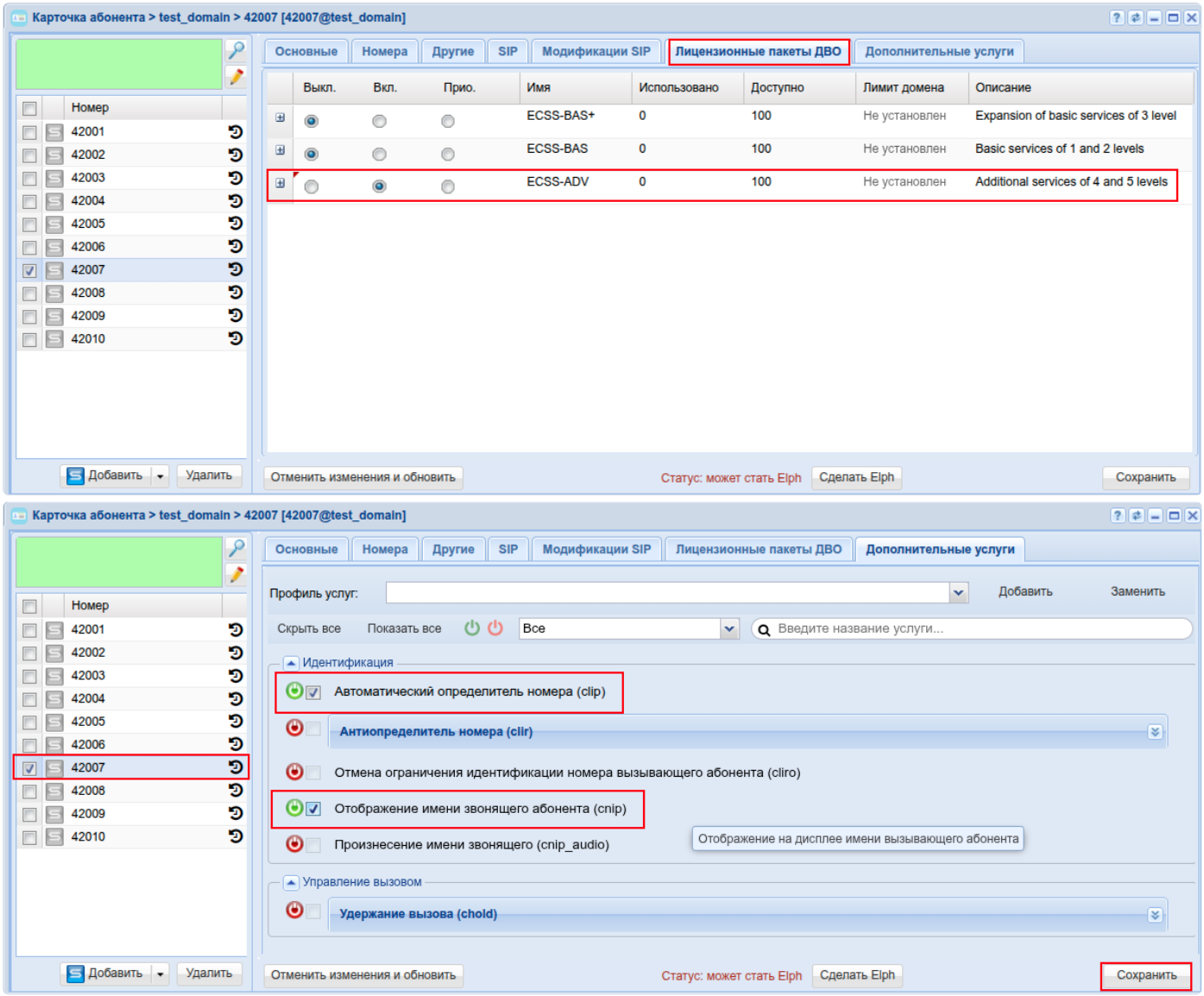


Рисунок 23 — Пример добавления услуг для абонента

11.6.8 Создание и применение контекстов маршрутизации для домена

Маршрутизация отвечает за поиск номера и последующую адресацию вызова. Для корректной работы системы должен быть настроен как минимум один контекст маршрутизации.

Настройка маршрутизации осуществляется в приложении "Менеджер маршрутизации" ("Routing manager") .

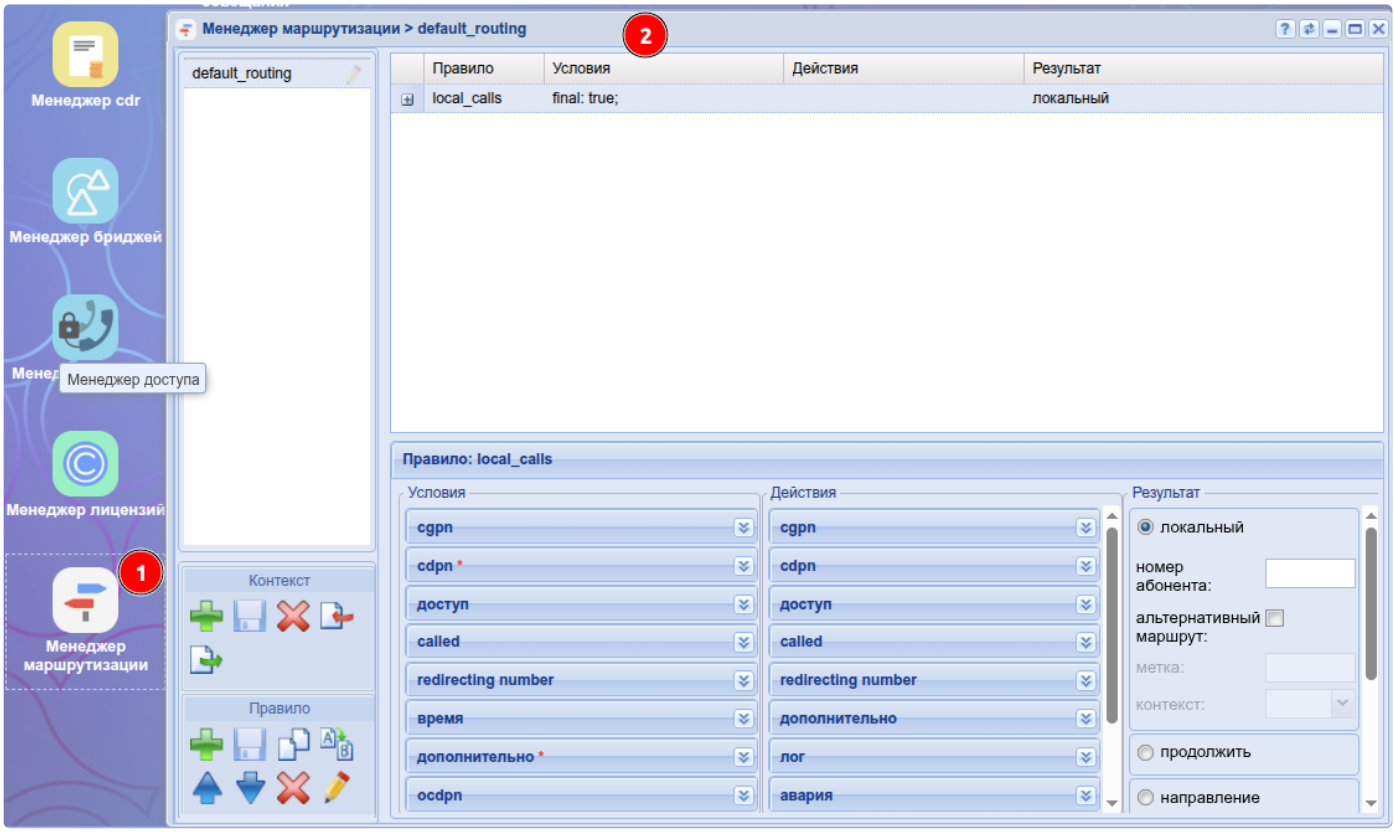


Рисунок 24 — Вид приложения "Менеджер маршрутизации"

В качестве примера создадим контекст, а затем несколько правил в нём, для этого:

1. В левой части окна в разделе "Контекст" нажмите на кнопку *"Создать контекст"* ;
2. В открывшемся диалоговом окне обозначьте имя контекста:

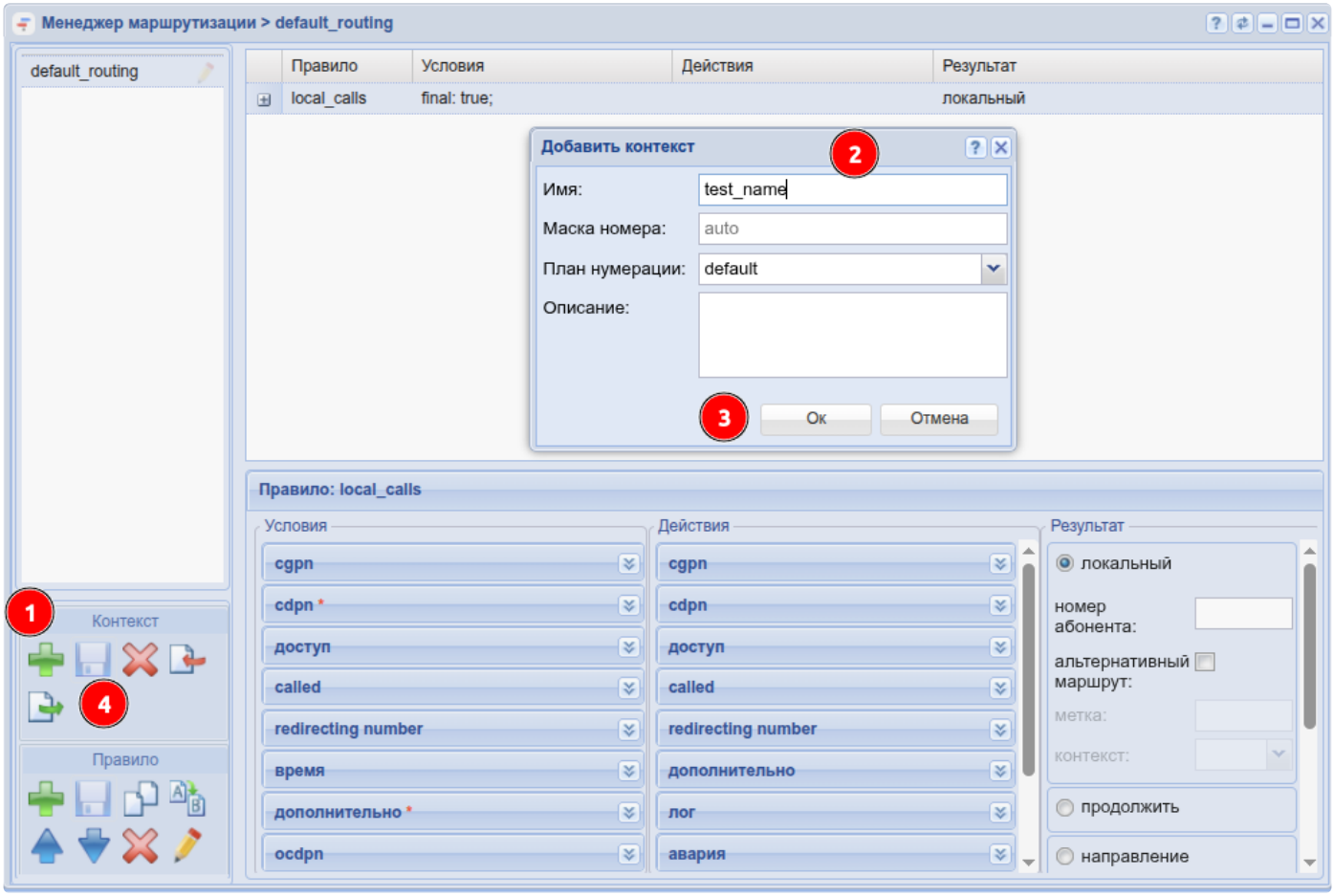


Рисунок 25 — Вид приложения "Менеджер маршрутизации" добавление нового контекста маршрутизации

3. Нажмите *"Сохранить контекст"*  ;

Создадим 4 правила в данном контексте:

- rule1 — правило для выхода на транк TAU-72;
- rule2 — правило локальной маршрутизации для номеров 42001-42005;
- rule3 — правило для выхода в ivr;
- rule4 — правило-исключение.

Для создания нового правила нужно выделить созданный контекст и нажать кнопку *"Создать правило"*. В появившемся окне нужно ввести название правила. Затем сохранить вновь созданные правила.

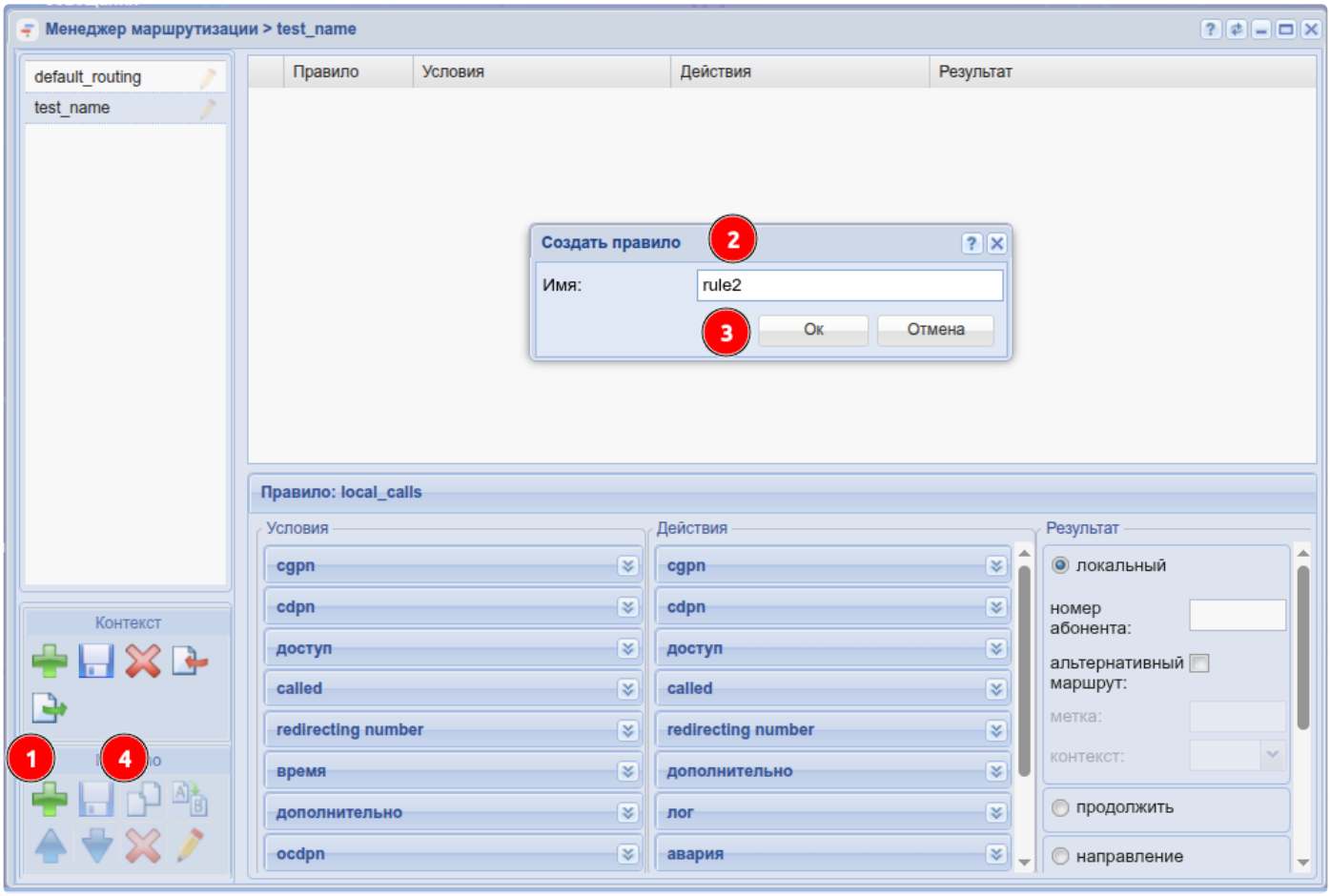


Рисунок 26 – Создание правила контекста маршрутизации

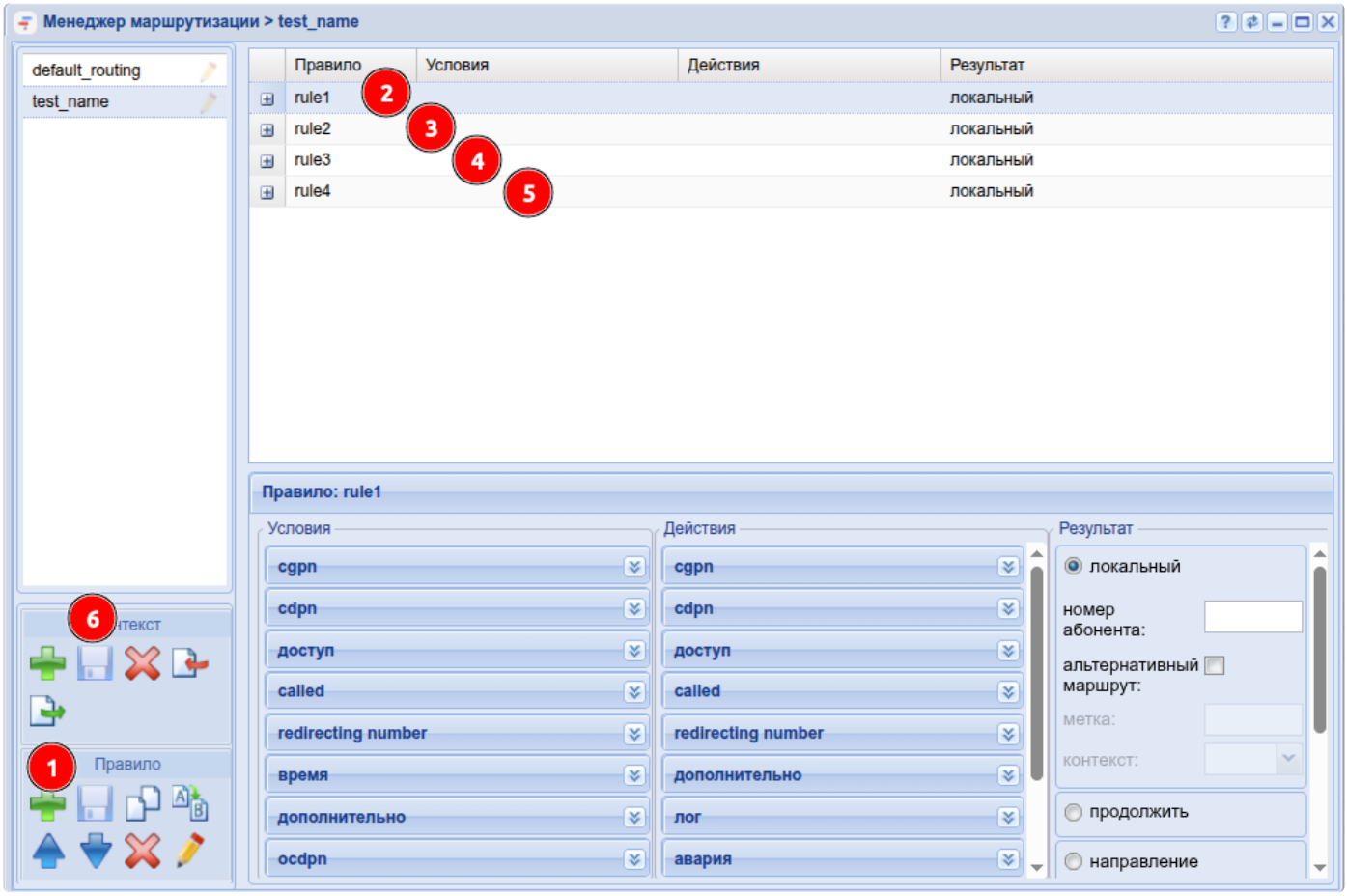


Рисунок 27 — Определение правил

На данный момент не определен транк, чтобы можно было сослаться на него в правиле, однако можно задать номера, по которым будет производится выборка.

Нажав на правило rule1, перейдем в нижнюю часть экрана, в которой располагаются области для редактирования контекста маршрутизации. Условно определим, что выборка для выхода на транк будет проводиться по признакам номера вызываемого абонента (CDPN), а номера в транке должны начинаться с цифры 4.

Функционально контекст маршрутизации делится на три части:

- Условия ("Condition") — раздел, определяющий выражения для выборки по предложенным признакам;
- Действия ("Action") — раздел, выполняющий преобразование признаков номеров к конкретному значению;
- Результат ("Result") — раздел, завершающий маршрутизацию и определяющий её результат.

rule1: Для выхода на транк нужно правильно отредактировать каждую часть:

- В разделе условий перейти на вкладку CDPN. Ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера с 42006 по 42007, нужно ввести условие: 420(6-7);
- В разделе действий перейти на вкладку CDPN. Ввести маску для изменения номера в поле "Номер". Например, чтобы добавить число 4 перед номером, нужно заполнить поле следующим выражением: 4{1,2,3,4,5};
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить транк в системе, поэтому к настройке данного правила вернемся чуть позже.

Таким же образом нужно настроить оставшиеся правила.

Для rule2:

- В разделе условий, во вкладке CDPN ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера со 42001 по 42005, ввести условие 420(1-5);
- В разделе результата определить результат как локальный (т.е. локальная маршрутизация);
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Для rule 3 допустим, что абоненты с номерами 108, 109 и 110 попадают в ivr-скрипт–информатор, прежде чем звонить далее.

- В разделе условий, во вкладке CGPN ввести номера телефонов, по которым будет выборка в поле "Номер", например 420(10,08-09).
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить в системе ivr-скрипт, поэтому к настройке данного правила вернемся чуть позже.

Для rule 4 мы определяем правило-исключение — это правило, которое работает в случае, если любые другие правила не срабатывают.

- По умолчанию данное правило создается в системе в контексте default_routing. В условиях на вызываемый номер прописывается символ %, а результатом является локальная маршрутизация, однако, если мы назначаем абоненту контекст, отличный от default_routing, в конце *рекомендуется* создавать это правило.

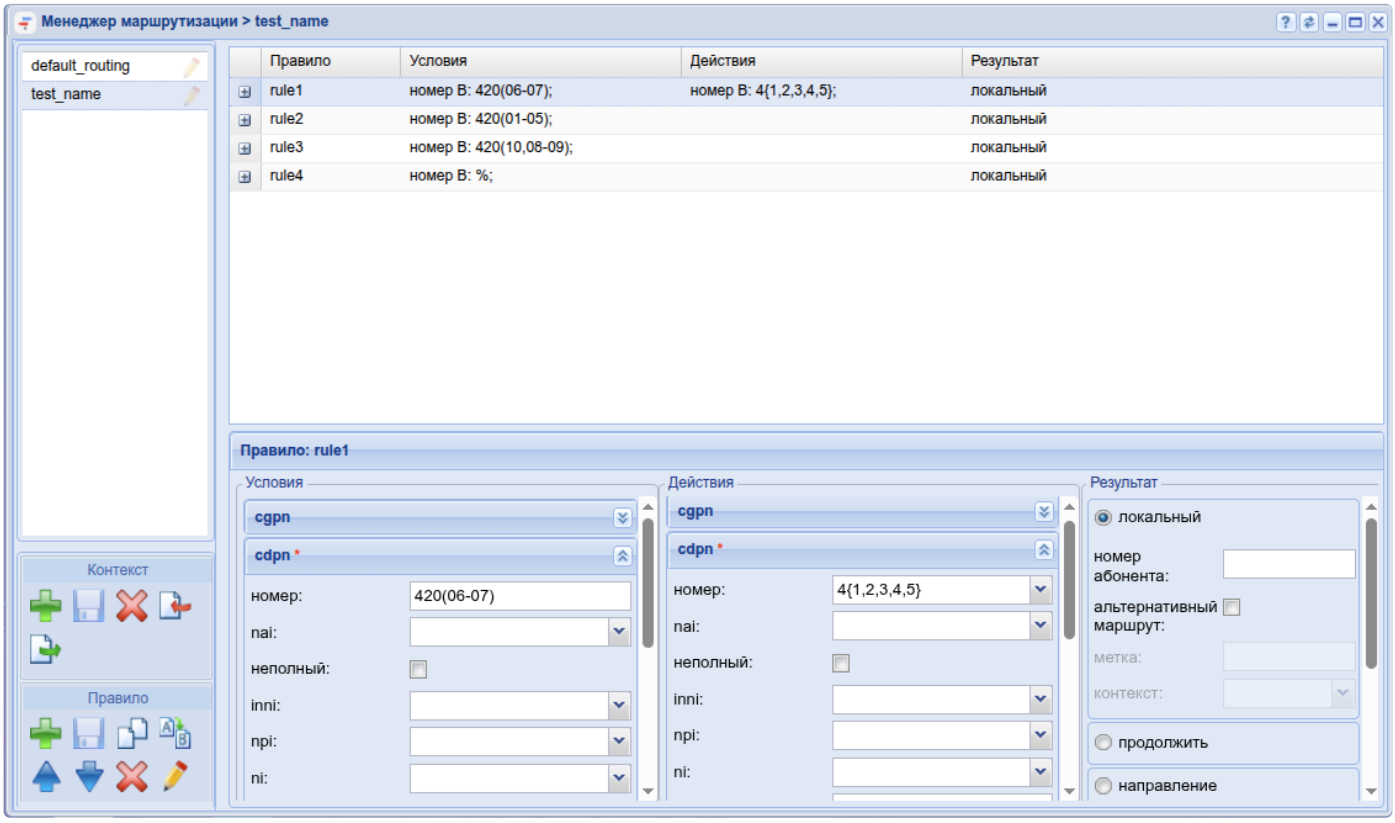


Рисунок 28 — Пример настроек контекста маршрутизации

11.6.9 Создание транка

Для создания и редактирования параметров транка в системе используется приложение "Менеджер транков" ("Trunk manager") .

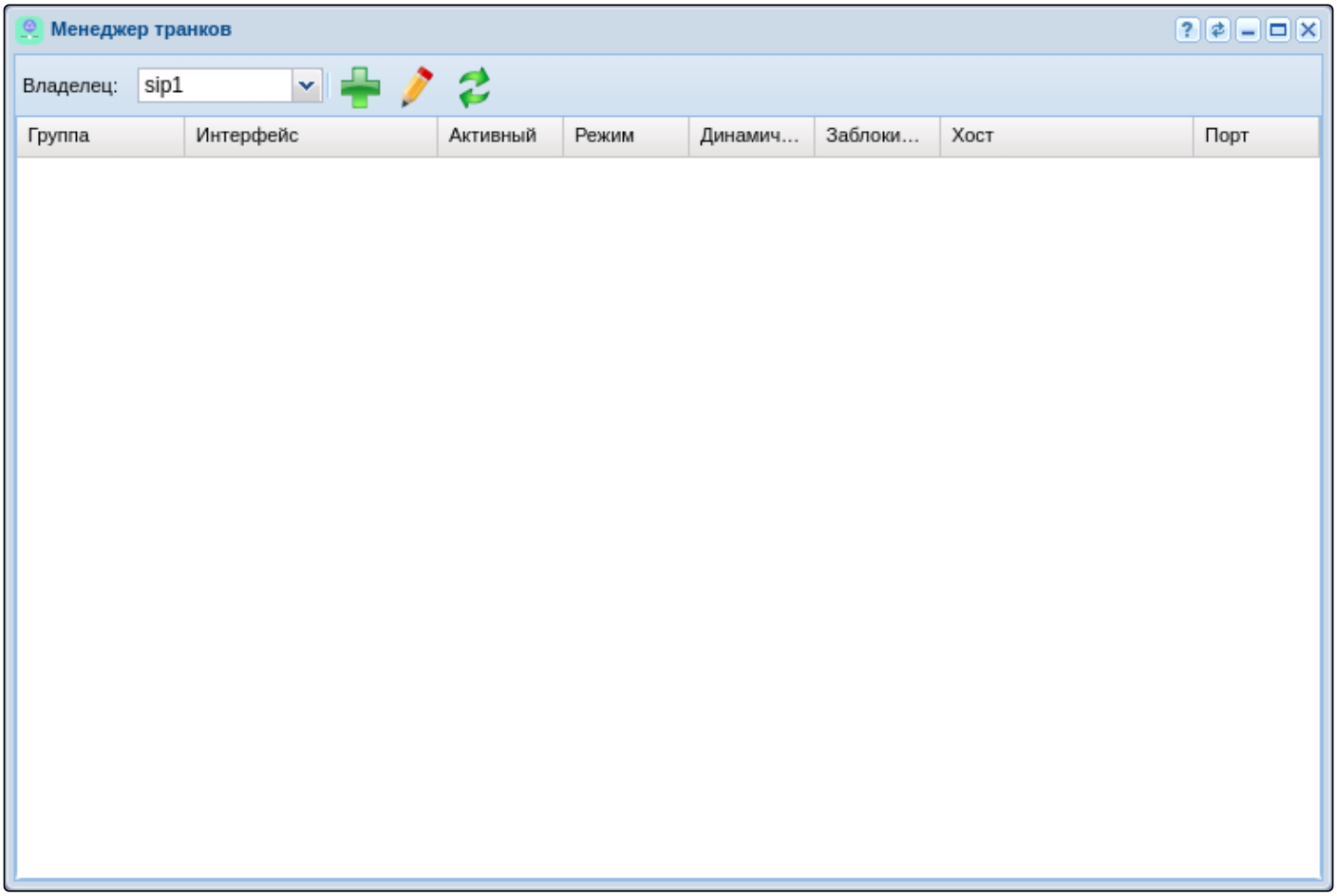


Рисунок 29 — Вид приложения "Менеджер транков"

Для определения транка в системе нужно нажать на кнопку *"Создать транк"* , в появившемся диалоговом окне определить параметры:

- *Имя* — назначить название транка, по которому его можно идентифицировать в системе;
- *Контекст* — применить ранее созданный контекст маршрутизации test_name;
- *Группа* — выбрать группу интерфейсов, созданную при определении абонентов "test.group";
- *Группа IP-адресов (IP-set)* — созданная на домене группа адресов "test_set";
- *Регистрация* — если используется транковая, то нужно включить данный параметр, в примере она не используется;
- *Хост [:порт]* — IP-адрес назначения транка — 10.0.3.100;
- *Порт для приёма* — транспортный порт, на котором будет слушаться трафик с транка, соответствует порту, назначенному на группу IP-адресов.

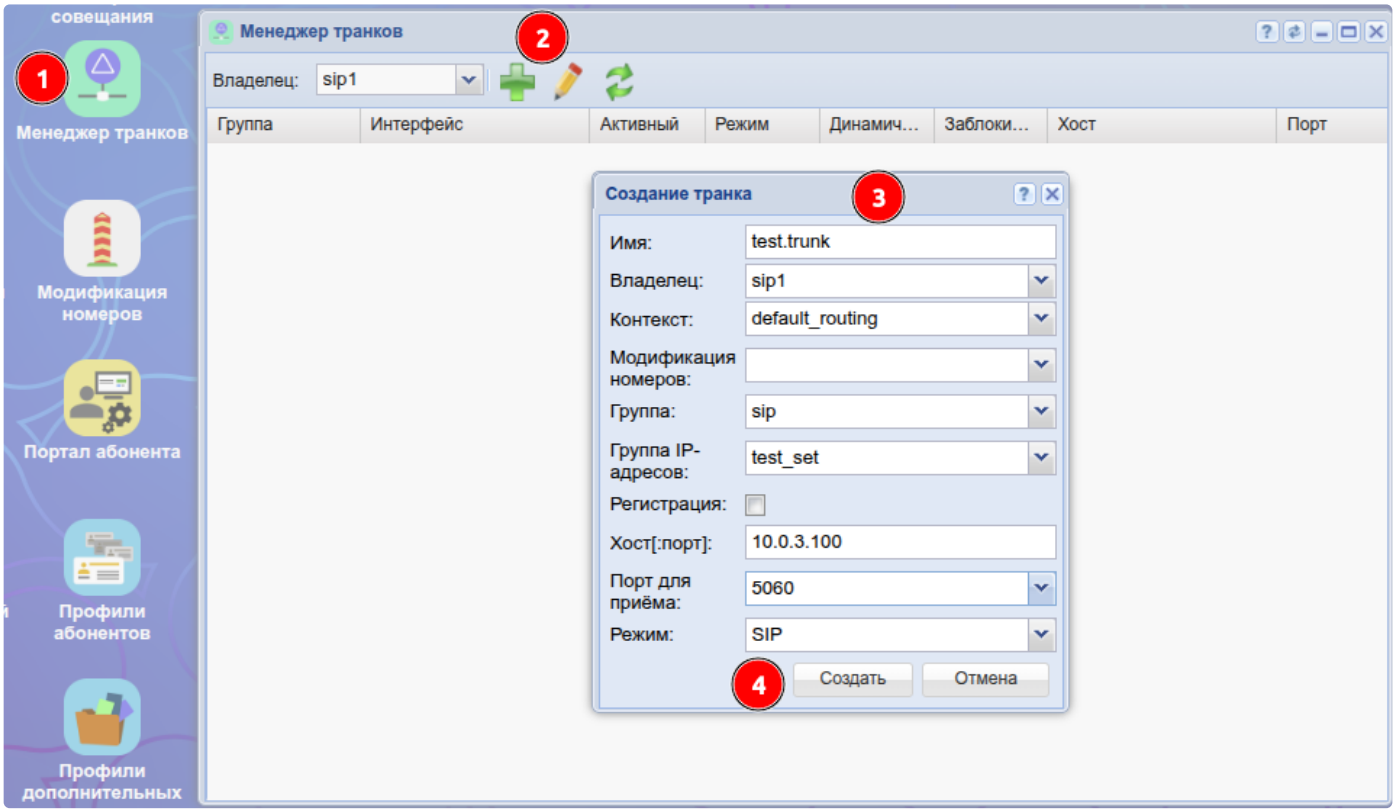


Рисунок 30 — Создание транка

11.6.10 Создание IVR-скрипта

Для создания IVR используется приложение "IVR-редактор" ("IVR editor").

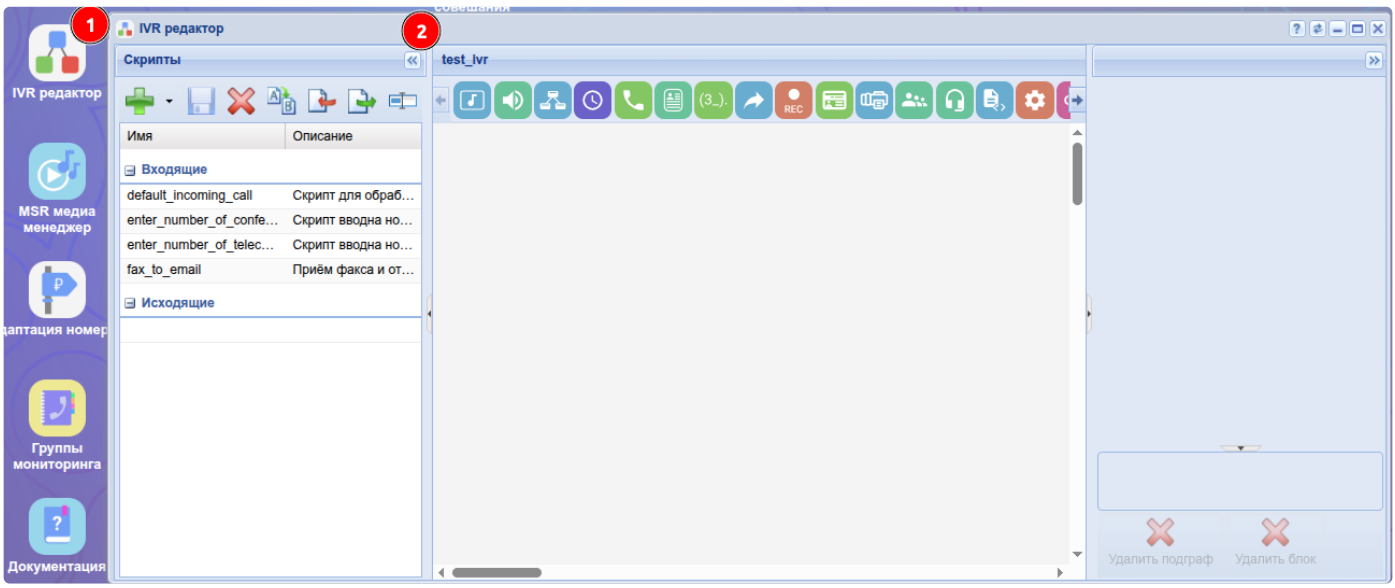


Рисунок 31 — Вид приложения "IVR-редактор"

Для создания скрипта нужно нажать на кнопку *"Добавить"* , выбрать тип скрипта (в данном случае будет скрипт на входящие вызовы), в диалоговом окне указать название скрипта, например "test_ivr".

После создания скрипта в основном окне редактора появится блок-схема.

В качестве примера составим скрипт, который при срабатывании проигрывает вызывающему абоненту заранее записанную фразу, а затем продолжает вызов.

Пример скрипта приведен на рисунках ниже. Здесь используются блоки Info и Dial.

- Блок Info — блок, который проигрывает сообщения до момента ответа пользователя. В качестве записанной фразы используется генератор тонов:

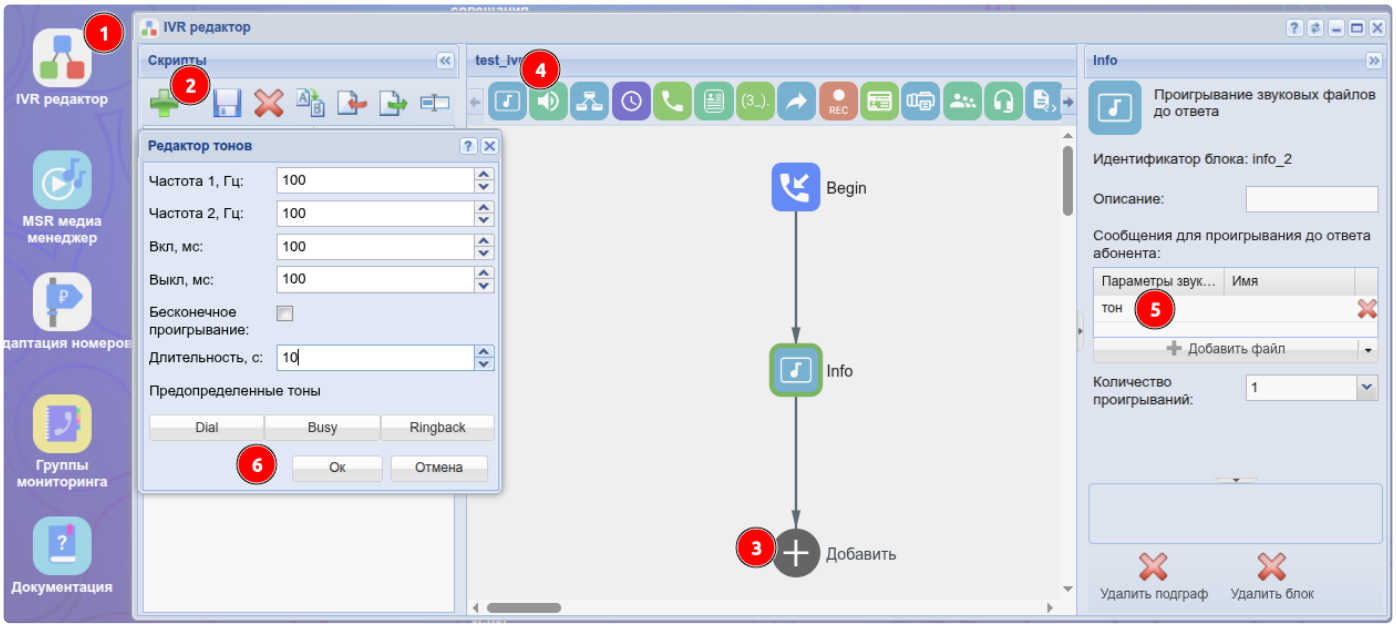


Рисунок 32 — Вид рабочей области IVR с настройками блока "Info"

- Блок dial — блок, который совершает вызов на заданный номер. Чтобы продолжить вызов, нужно применить предопределенную переменную CDPN:

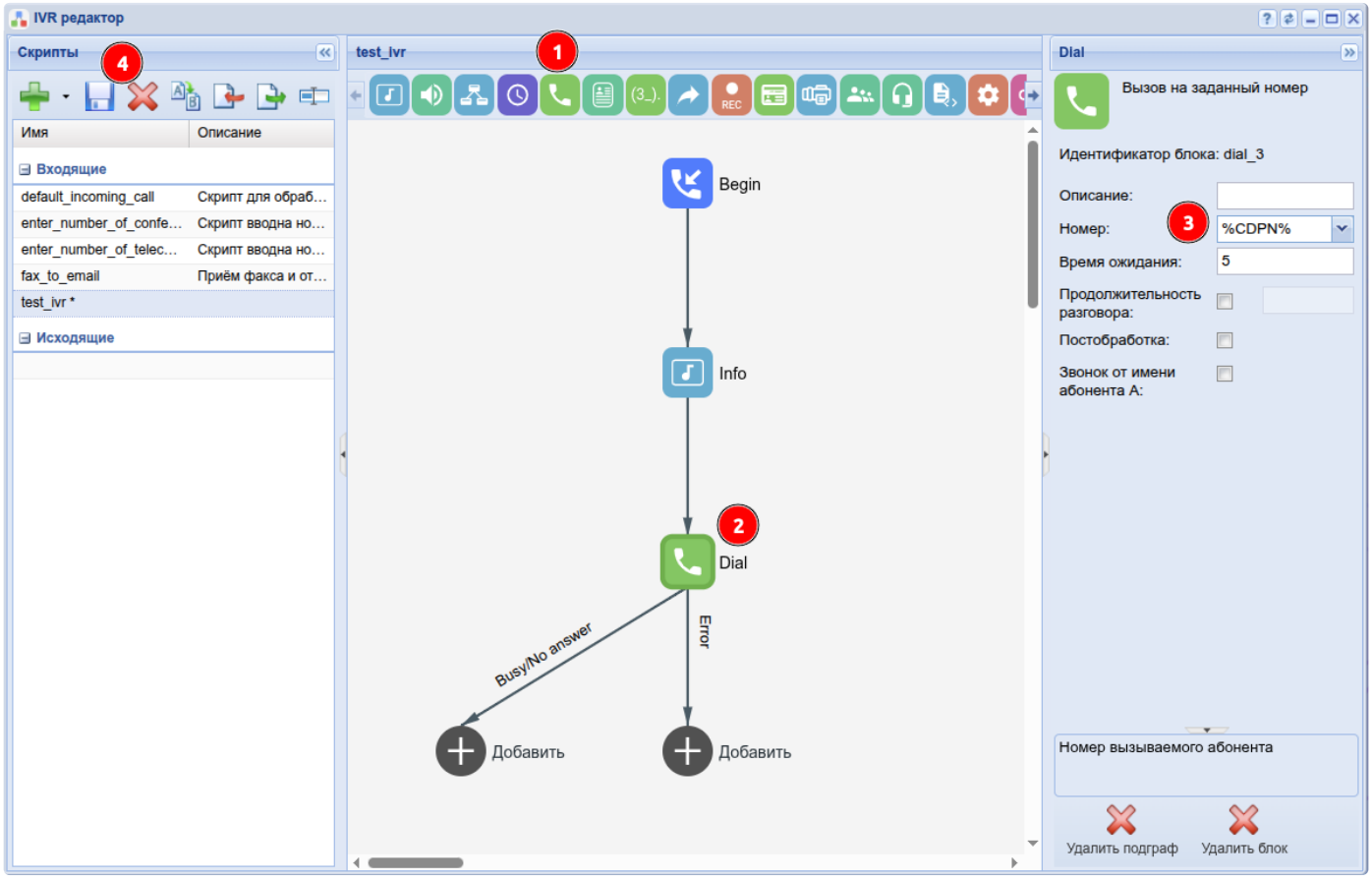


Рисунок 33 — Вид рабочей области IVR

11.6.11 Завершение настройки маршрутизации

Для завершения настройки маршрутизации нужно войти в приложение "Менеджер маршрутизации" ("Routing manager") и в правилах rule1 и rule3 настроить соответствующие результаты маршрутизации, для этого:

- rule1: в разделе "Результат" выберите подпункт "внешний". В таблице "Направления" в поле "Значение" добавьте созданный ранее транк, после чего сохраните правило .
- rule3: В разделе "Результат" выберите подпункт "ivr" и в поле скрипт добавьте созданный ранее ivr-скрипт, после чего сохраните правило и контекст .

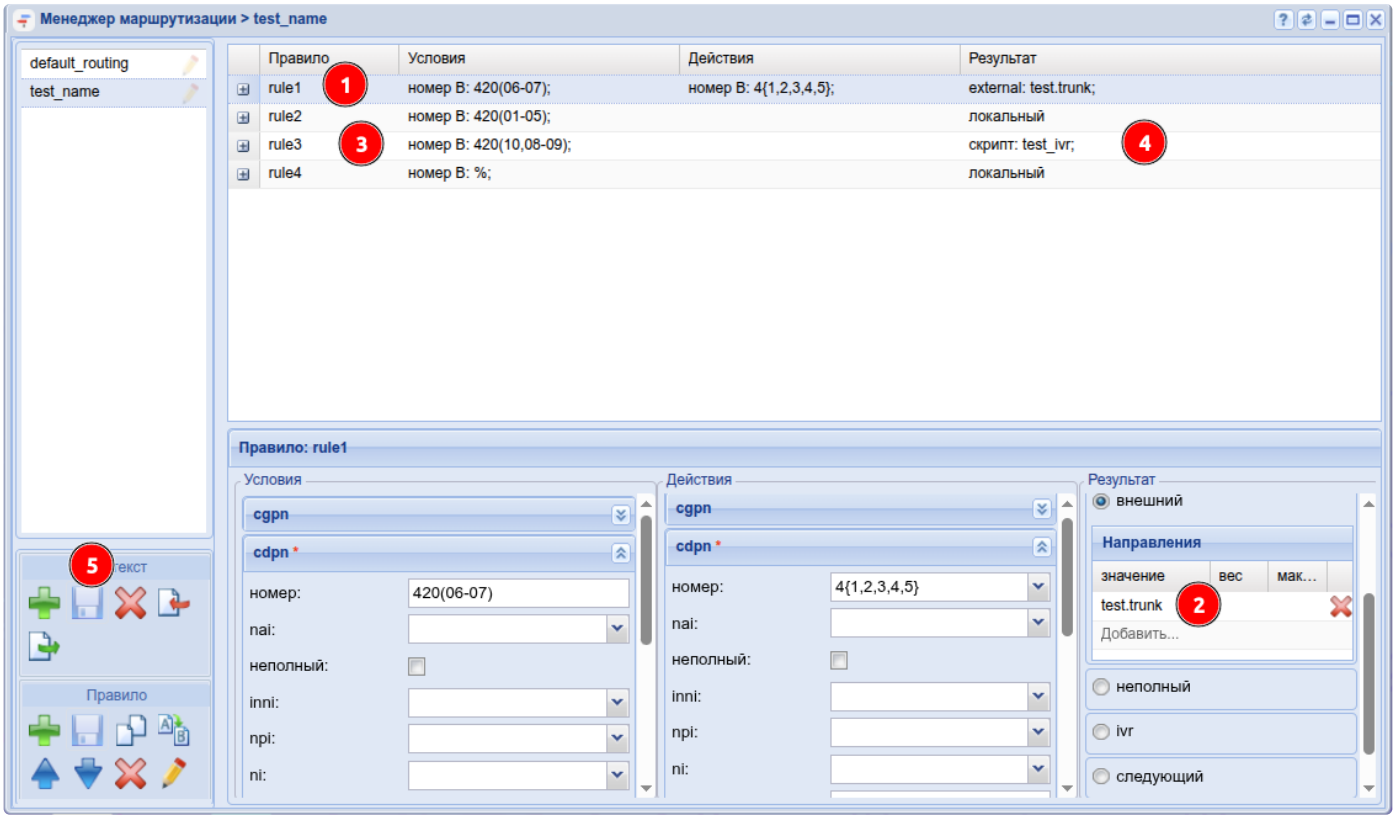


Рисунок 34 — Вид контекста маршрутизации

11.7 Первоначальная конфигурация системы через интерфейс командной строки (CLI)

пример первоначальной конфигурации через CoCon (CLI)

11.7.1 Добавление медиа-ресурсов в интерфейсе командной строки (CLI)

⚠ Для возможности использования медиаресурса и установления соединений , необходимо выполнить активизацию в команде создания медиаресурса - **/system/media/resource/declare** (параметр **"is active"=true**) или **/system/media/resource/set** (параметр **"active"=true**), если при создании активизация не была выполнена. Так же можно выполнить активизацию через веб интерфейс , приложение - сетевые окончания MSR (MSR-registrars).

Зайдите в CLI - CoCon:

```
ssh admin@localhost -p 8023
password: password
```

Медиаресурс — это описание параметров медиасервера, необходимых для работы с ним.

Медиаресурс — это описание параметров медиасервера, необходимых для работы с ним.

declare
Команда для объявления подключения медиасервера к ECSS-10.

Для работы в режиме предварительно настроенной системы можно заранее прописать медиаресурсы, информация о которых известна.

Для этого необходимо создать описание известных медиасерверов с указанием всех параметров, что позволить начать работу с медиасервером сразу после его регистрации.

Команда для создания конфигурации медиасервера:

Путь команды:

/system/media/resource/declare

Синтаксис:

declare <node> by_contact <contact> <zone> <site> <is active>

или

declare <node> by_iface <msr> <acc_id> <iface> <zone> <site> <is active>

Параметры:

<node> — имя ноды core;
<contact> — contact uri декларируемого медиаресурса;
<zone> — логическая зона обслуживания MSR;
<site> — идентификатор сети обслуживания MSR;
<is active> — состояние активности медиаресурса, значения true/false;
<msr> — имя MSR;
<acc_id> — номер идентификатор ресурса;
<iface> — название интерфейса.

Установить связь MSR- и Core-подсистем. Для этого выполнить команды:

```
/system/media/resource/declare core1@ecss1 contact net.20@msr.ecss1  default local true
```

результат

```
/system/media/resource/info core1@ecss1 net.20@msr.ecss1

  Media resource specific for node core1@ecss1:
┌─┬──────────────────┬──────────────────────┐
│F│    Parameter     │        Value         │
├─┼──────────────────┼──────────────────────┤
│r│ Contact          │ net.20@msr.ecss1     │
```

```
|r| Status           | registered           |
|r| Cc-status        | connected            |
|r| Cc-status uptime| 00:29:06             |
|r| Cc-id            | 63bfc9               |
|r| Address          | 10.0.20.40:5700      |
|r| Register time    | 2025/12/02 13:38:58 |
|r| Expired          | 113                  |
|r| Iface            | net.20(10.0.20.40)   |
| | Active           | true                 |
| | Zone             | default              |
| | Site             | local                |
|r| MSR version      | 3.18.1.0.10          |
|r| MSR load         | 0                    |
|r| MSR perf coef    | 1.0                  |
└─┴──────────────────┴──────────────────────┘

Legend:
Column F - means Flag, values r - flag for readonly parameter
```

результат / список

```
/system/media/resource/list all
  All media resource selected list specific:
┌──────────────┬──────────┬──────────────┬──────────┬──────┬────────┬──────────┬──────────
──┬──────────┬──────────┐
│    Node      │    MSR   │     MSR      │    MSR   │ MSR  │ Cc-id  │ Cc-status │ Cc-uptime
│   Cc-address │ Iface    │   Iface      │ Active   │ Zone │ Site   │    Contact    │
   Status  │ Expired │
│              │          │   version    │ perf coef │ load │        │           │
│              │  name    │    addr      │          │      │        │           │
│         │
├──────────────┼──────────┼──────────────┼──────────┼──────┼────────┼──────────┼──────────
──┼──────────┼──────────┤
│ core1@ecss1 │ msr.ecss1 │ 3.18.1.0.10 │      1.0 │    0 │ 63bfc9 │ connected │ 00:27:50
│ 10.0.20.40:5700 │ net.20 │ 10.0.20.40 │ true    │ default │ local │ net.20@msr.ecss1 │
 registered │      68 │
└──────────────┴──────────┴──────────────┴──────────┴──────┴────────┴──────────┴──────────
──┴──────────┴──────────┘
```

11.7.2 Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI)

/system/media/registrar/ - команды управления регистратором для MSR медиа менеджера

/system/media/registrar/ - команды управления регистратором для MSR медиа менеджера

set

Команда для изменения общих и индивидуальных параметров.

Путь команды:

/system/media/registrar/set

Синтаксис настройки общих параметров:

set common <FIELD> <VALUE>

Параметры:

<FIELD> - имя настраиваемого параметра;

<VALUE> - значение настраиваемого параметра

Где

- common - тип выставляемого параметра;
- <common parameter> - общий параметр медиарегистратора для всех нод: registration-timeout;
- private - тип выставляемого параметра;
- <node name> - имя ноды на которой находится медиарегистратор;
- <private parameter> - параметр медиарегистратора, который выставляется на <node name>: is-active, listen-interface & listen-port.

Список параметров доступных для настройки:

- is-active - позволяет включить или выключить регистратор (false - деактивирует регистратор, регистратор перестает принимать SIP трафик);
- listen-interface - имя интерфейса, который использует регистратор;
- listen-port - порт, на котором регистратор принимает запросы на регистрацию (по умолчанию 5000);
- registration-timeout - таймер перерегистрации, диапазон значений от 60 до 86400.

В общих параметрах доступна настройка времени регистрации, по умолчанию установлено 60 секунд.

Пример:
Установить время регистрации 60 секунд:

```
/system/media/registrar/set common registration-timeout 60
Common parameter registration-timeout set to 60.
```

Синтаксис настройки индивидуальных параметров:

system/media/registrar/set private <CORE> <FIELD> <VALUE>

Параметры:

<CORE> - нода ядра, для которой настраивается Registrar-сервер;

<FIELD> - имя настраиваемого параметра;

<VALUE> - значение настраиваемого параметра.

Настраиваемые параметры:

- is-active - активен ли Registrar-сервер, принимает значения true | false;
- listen-ip - ip-адрес, на котором Registrar-сервер будет принимать соединения (по умолчанию 0.0.0.0);
- listen-port - сетевой порт, на котором Registrar-сервер будет принимать соединения (по умолчанию 5000).

Установить ipадрес для приема соединений от медиасервера:

```
/system/media/registrar/set private core1@ecss1 listen-ip 10.0.20.40
Private parameter listen-ip set to 10.0.20.40 at node core1@ecss1.
```

Результат:

```
/system/media/registrar/info
 Common parameters:
┌──────────────────────────────┬─────────┐
│          Parameter           │  Value  │
├──────────────────────────────┼─────────┤
│ registration-timeout, sec    │   180   │
└──────────────────────────────┴─────────┘
 Private parameters:
┌──────────────┬──────────┬────────────┬─────────────┬──────────────┐
│     Node     │  Status  │ Is-active  │  Listen-ip  │ Listen-port  │
├──────────────┼──────────┼────────────┼─────────────┼──────────────┤
│ core1@ecss1  │ active   │ true       │ 10.0.20.40  │        5000  │
└──────────────┴──────────┴────────────┴─────────────┴──────────────┘
```

После того как все сервисы будут запущены, ноды какое-то время будут устанавливать связь. Как только все ноды загрузятся , при выполнении запроса **system-status**, последует вывод следующей информации:

```
/system-status
Checking...
┌─┬──────────────┬──────────────────────────────┬──────────────┬──────────┬──────┐
│ │     Node     │           Release            │ Erlang nodes │Mnesia nodes│Uptime │
├─┼──────────────┼──────────────────────────────┼──────────────┼──────────┼──────┤
│ │core1@ecss1   │ecss-core-3.18.1.0.1655       │core1@ecss1   │not running │31m 25s│
│ │ds1@ecss1     │ecss-ds-3.18.1.0.1655         │ds1@ecss1     │ds1@ecss1   │31m 25s│
│ │md1@ecss1     │ecss-mediator-3.18.1.0.1655   │md1@ecss1     │md1@ecss1   │31m 26s│
│ │mycelium1@ecss1│ecss-mycelium-3.18.1.0.1655  │mycelium1@ecss1│not running │31m 26s│
│ │sip1@ecss1    │ecss-pa-sip-3.18.1.0.1655     │sip1@ecss1    │sip1@ecss1  │31m 26s│
└─┴──────────────┴──────────────────────────────┴──────────────┴──────────┴──────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────┬────────────┬─────────────┐
│     Node     │   MSR    │   MSR    │ Cc-status  │ Cc-uptime   │
│              │          │ version  │            │             │
├──────────────┼──────────┼──────────┼────────────┼─────────────┤
│ core1@ecss1  │ msr.ecss1 │ 3.18.1.0.10 │ connected │ 00:29:51  │
└──────────────┴──────────┴──────────┴────────────┴─────────────┘
```

В примере выше видно, что ноды вошли в кластер и *MSR* зарегистрировались на ноде *ecss-core*.

Создание и настройка группы SIP-IP-адресов (IP-set)

IP-set – совокупность одного или нескольких IP-адресов из одной подсети и списка слушающих UDP/TCP-портов, которые будут открыты SIP-адаптером на данных адресах.

Произведите настройку адаптера SIP согласно ТЗ (команды выполнить в CoCon):

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss1 ip =
10.0.20.40
Property "ip_set" successfully changed from:

to
test_set: no ports set
test_set: sip1@ecss1 10.0.20.40
test_set: dscp 0.
```

```
/cluster/adapter/sip1/sip/network/set ip_set test_set listen-ports list = [5060]
Property "ip_set" successfully changed from:
test_set: no ports set
test_set: sip1@ecss1 10.0.20.40
test_set: dscp 0
   to
test_set: 5060
test_set: sip1@ecss1 10.0.20.40
test_set: dscp 0
```

11.7.3 Создание Домена и подключение IP-set

Создайте Домен (виртуальную АТС) и присвойте ему созданную выше группу (IP-set) настроек SIP-адаптера (команды выполняются в CoCon):

```
/domain/declare test_domain --add-domain-admin-privileges --add-domain-user-privileges
New domain test_domain is declared
```

```
/domain/test_domain/sip/network/set ip_set [test_set]

Property "ip_set" successfully changed from:

[]
    to
["test_set"].
```

После создания домена, администратор системы должен открыть доступ на управление услугами оператору Домена (виртуальной АТС). Для этого используются команду:

Разрешить оператору Домена (виртуальной АТС) "test_domain" управлять всеми услугами :

```
/cluster/storage/ds1/ss/access-list add test_domain *
```

```
Supplementary services successfully added in the domain test_domain.
```

Для возможности применения пакета услуг у абонентов сразу после создания, необходимо настроить политику применения лицензионных пакетов услуг (детальную информацию вы можете посмотреть здесь)

Создать политику применения лицензии с именем -"new_sub"

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/declare new_sub "Задаются при создании абонента"
```

Задать имя пакета услуг дополнительных сервисов, для этого использовать следующие команды :

выполняется в CoCon user/pw admin/password

```
/cluster/storage/ds1/licence/show-licence 1
```

нас интересует следующая информация :

```
/cluster/storage/ds1/licence/show-licence 1
SSW ID                                                  < Имя SSW ID >
Description                                             Лицензия для тестирования.

. . .

Supplementary Services licence package(s)
 name                                                   'ECSS-ADV'
 limit                                                  100
 description                                            "Additional services of 4 and
5 levels"
 SS list                                                [1,2,3,4,5]

. . .
```

имя пакета услуг в примере "ECSS-ADV".
Добавить в созданную политику пакет лицензионных услуг с именем "ECSS-ADV", именно так он описан в лицензии:

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/package-add new_sub ECSS-ADV
```

Пакет добавлен, но в настоящий момент политика не активна (не может быть применена к абоненту). Для активации политики new_sub выполнить следующую команду:

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/activate new_sub
```

Теперь при создании абонентов можем использовать дополнительные услуги.

12 Установка ECSS10 3.18 на Astra Linux, в закрытом контуре, установка в кластере

12.1 Первоначальная установка системы с резервированием, в кластере из двух серверов

12.1.1 Исходные данные

Вариант с резервом, используется на проектах чаще , поэтому его мы рассмотрим первым. Пример установки на одном сервере

Конфигурация системы в кластере из двух серверов

Исходные данные

В данном разделе использованы специфичные для данного продукта термины, ознакомиться с ними можно здесь.
Техническое задание:
Требуется интеграция программного коммутатора 5 класса ECSS-10 (SSW) на 2-х физических серверах с поддержкой SIP со следующими параметрами на нагрузку группы серверов:

- Максимальное количество абонентов 15000 (MUL — Max user limit);
- Максимальное количество одновременных соединений 2000 (MCL — Max call limit);
- Требуется полный резерв системы ecss (все node резервируется на втором сервере);
- Количество сетевых интерфейсов ethernet — 3/5.

По техническому заданию требуется определить аппаратную платформу.

Полоса пропускания СПД

Требуемая полоса пропускания СПД **не менее** 1000Mb/s

Пример составления аппаратных требований:

Пример составления аппаратных требований:

Пример разнесения составляющих по адресному пространству				
Имя сервера (хоста)	**Роль**	**Интерфейс**	**Адрес/Маска**	**Порт**
Статические адреса программного коммутатора				
ecss1	Интерфейс управления сервером (порт 2000 ssh)	eth0	192.168.56.11/24	2000
ecss2	Интерфейс управления сервером (порт 2000 ssh)	eth0	192.168.56.12/24	2000

ecss1	Адрес ядра (ecss-core)	net.20	10.0.20.11/24	5000
ecss2	Адрес ядра (ecss-core)	net.20	10.0.20.12/24	5000
ecss1	Адрес OAM	net.10	10.0.10.41/24	-
ecss2	Адрес OAM	net.10	10.0.10.42/24	-
ecss1, ecss2	Адрес шлюза	eth0	192.168.56.1	-
ecss1, ecss2	Адреса DNS-серверов	eth0	8.8.8.8	-
ecss1, ecss2	Адреса NTP-серверов	eth0	10.136.16.211, 10.136.16.212	123
Виртуальные адреса программного коммутатора				
ecss1	Виртуальный адрес программного адаптера хоста ecss1	net.20:SIP1	10.0.20.31/24	-
ecss2	Виртуальный адрес программного адаптера хоста ecss2	net.20:SIP2	10.0.20.32/24	-
ecss1	Запасной виртуальный адрес программного адаптера хоста ecss2 на хосте ecss1	net.20:SIP2	10.0.20.32/24	-
ecss2	Запасной виртуальный адрес программного адаптера хоста ecss1 на хосте ecss2	net.20:SIP1	10.0.20.31/24	-
Заполнить данные параметры для SSW из проектного технического задания				
Имя сервера (хоста)	**Роль**	**Интерфейс**	**Адрес/Маска**	**Порт**
Статические адреса программного коммутатора				
ecss1	Интерфейс управления сервером (порт 2000 ssh)			2000
ecss2	Интерфейс управления сервером (порт 2000 ssh)			2000
ecss1	Адрес ядра (ecss-core)			5000

ecss2	Адрес ядра (ecss-core)			5000
ecss1, ecss2	Адрес шлюза			-
ecss1, ecss2	Адреса DNS-серверов			-
ecss1, ecss2	Адреса NTP-серверов			123
Виртуальные адреса программного коммутатора				
ecss1	Виртуальный адрес программного адаптера хоста ecss1			-
ecss2	Виртуальный адрес программного адаптера хоста ecss2			-
ecss1	Запасной виртуальный адрес программного адаптера хоста ecss2 на хосте ecss1			-
ecss2	Запасной виртуальный адрес программного адаптера хоста ecss1 на хосте ecss2			-

Включение в сеть

Топологию включения сервера в сеть для обеспечения резервирования рекомендуется делать с использованием 2-х коммутаторов. Пример на рисунке ниже.

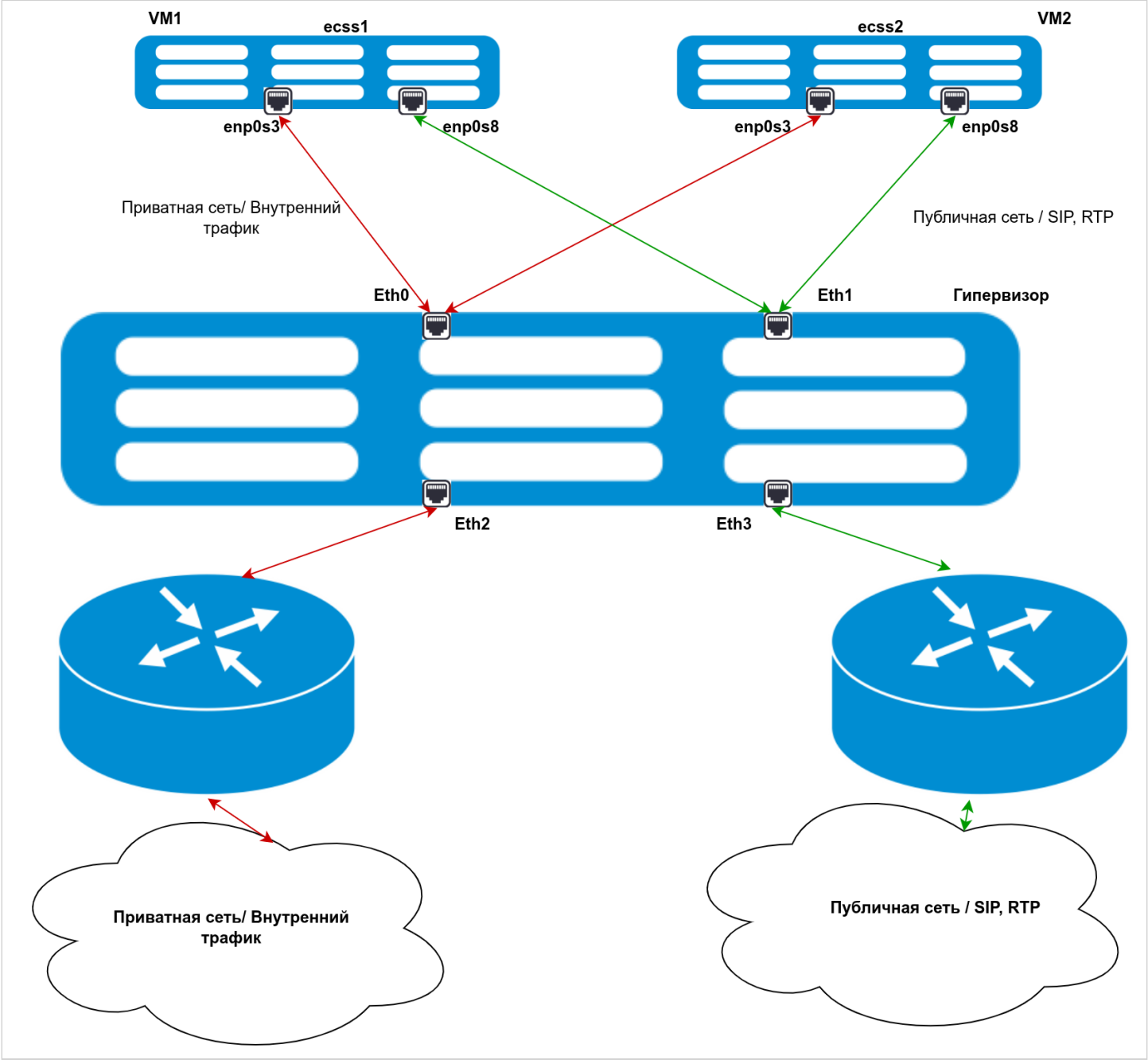


Рисунок 1 — Схема организации связи

Вариант 1. Схема active-backup

Коммутаторы связаны между собой в erps-кольцо.

Все 4 физических сетевых интерфейса каждого сервера объединяются в 1 агрегированный линк (bond). Агрегация портов сервера настраивается в режиме **active-backup**, т.е. в работе всегда находится только 1 сетевой интерфейс. Сетевые интерфейсы каждого сервера попарно включены в коммутаторы, на которых агрегация портов (port-channel) также настраивается в режиме active-backup. Например, eth1 и eth2 каждого сервера включаются в первый коммутатор (port-channel 1 и 2), а eth3 и eth4 (port-channel 3 и 4) — во второй.

Вариант 2. Схема LACP

Коммутаторы связаны между собой в стек. Стек должен логически работать как один коммутатор, способный обеспечивать агрегацию портов в режиме LACP между разными физическими коммутаторами. Пример — коммутаторы MES-3124 со специализированной прошивкой.

Все 4 физических сетевых интерфейса также объединяются в 1 агрегированный линк (bond). Агрегация портов сервера настраивается в режиме **802.3ad**. Создаются агрегированные группы сетевых карт с одинаковой скоростью и дуплексом. При таком объединении передача задействует все каналы в активной агрегации согласно стандарту IEEE 802.3ad. Выбор того, через какой интерфейс необходимо отправлять пакет, определяется политикой. По умолчанию это XOR-политика, можно использовать «xmit_hash» политику. Подробнее — в разделе Приложение E. Netplan.

Требования:

- Поддержка Ethtool в драйвере для получения информации о скорости и дуплексе на каждом сетевом интерфейсе;
- Поддержка на коммутаторе стандарта IEEE 802.3ad;

Сетевые интерфейсы сервера также попарно включены в коммутаторы, на которых агрегация портов (port-channel) тоже настраивается в режиме LACP. Например, eth1 и eth2 каждого сервера включаются в первый коммутатор (port-channel 1 и 2), а eth3 и eth4 (port-channel 3 и 4) — во второй.

12.1.2 Последовательность установки SSW ECSS10 в кластере из двух серверов

Установка SSW ECSS10 состоит из 2-х основных частей :

- Подготовительной - Установка ОС Astra Linux 1.7, обновлений пакетов ОС Astra Linux 1.7 , установка дополнительных пакетов программ, подготовка сетевых интерфейсов, /etc/hosts, ssh-keygen
- Установка приложений Eltex ECSS10

Подготовительный этап

В этом разделе приведено описание инсталляции операционной системы, а также необходимых и дополнительных пакетов. Система ECSS-10 версии 3.14 работает под управлением ОС Astra Linux 1.7.

Предварительные требования

- Установочный загрузочный носитель с дистрибутивом операционной системы;
- Подготовленный сервер с обновленным BIOS, ILO (если есть), подключенная сеть для доступа;
- Выставленный первый приоритет загрузки с установочного носителя — USB Flash или CD/DVD в BIOS;
- Достаточный объем дискового пространства и памяти в соответствии с проектом.

12.1.3 Установка ОС

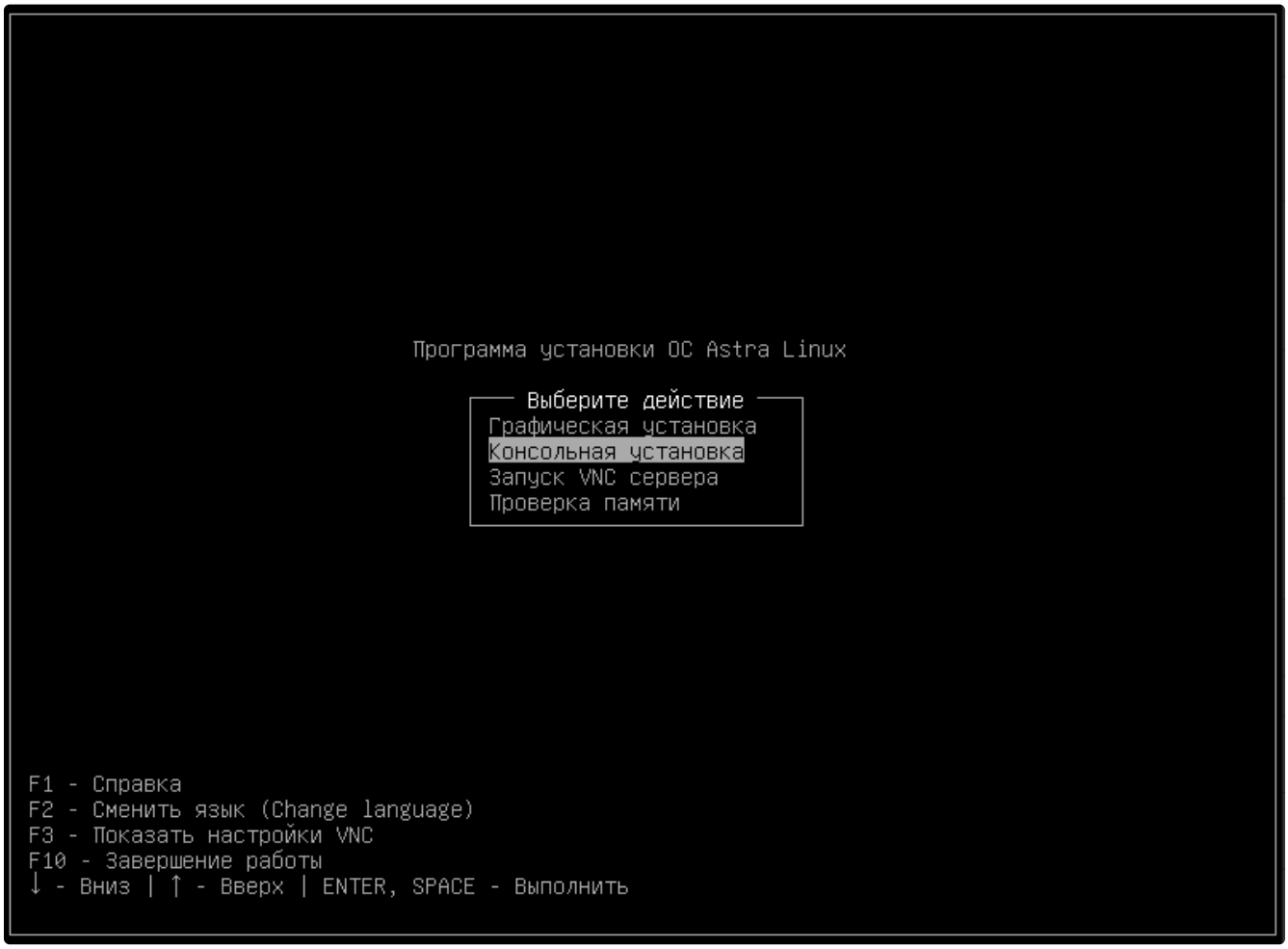
Для установки ОС необходимо выполнить следующее:

- После загрузки с установочного носителя выбрать "Графическая установка" или "Установка".
- Согласится с лицензионным соглашением.
- Выбрать раскладку клавиатуры.
- Выбрать сетевой интерфейс для подключения к сети Интернет.
- Установить имя компьютера (рекомендованы ecss1/ecss2)
- Настроить пользователя- администратора.
- Создать партиции на диске в соответствии с таблицей 1 (разметка диска - Вручную).
- В процессе установке , **не создавать раздел swap** . При напоминании о пропуске раздела , **подтвердить** , что установка выполняется без создания swap.
- Установить часовой пояс.

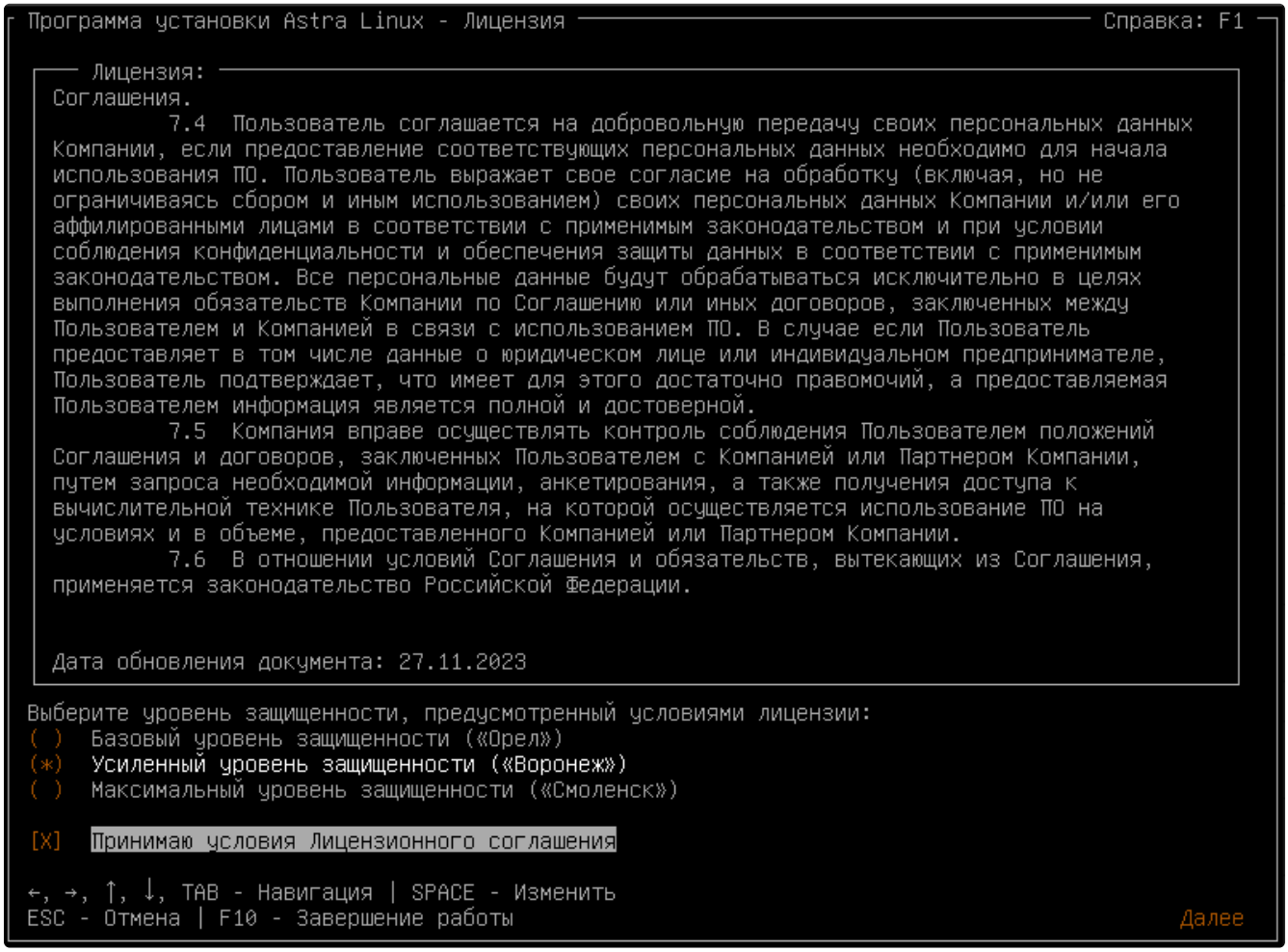
- Установить OpenSSH server (активизировать пункт : "Средства удаленного подключения SSH").
- Выбрать уровень защищенности, который используется в вашем проекте
- Дополнительные настройки ОС - по умолчанию
- Перезагрузить успешно установленную ОС

Установка ОС Astra Linux 1.8

Выбрать режим установки



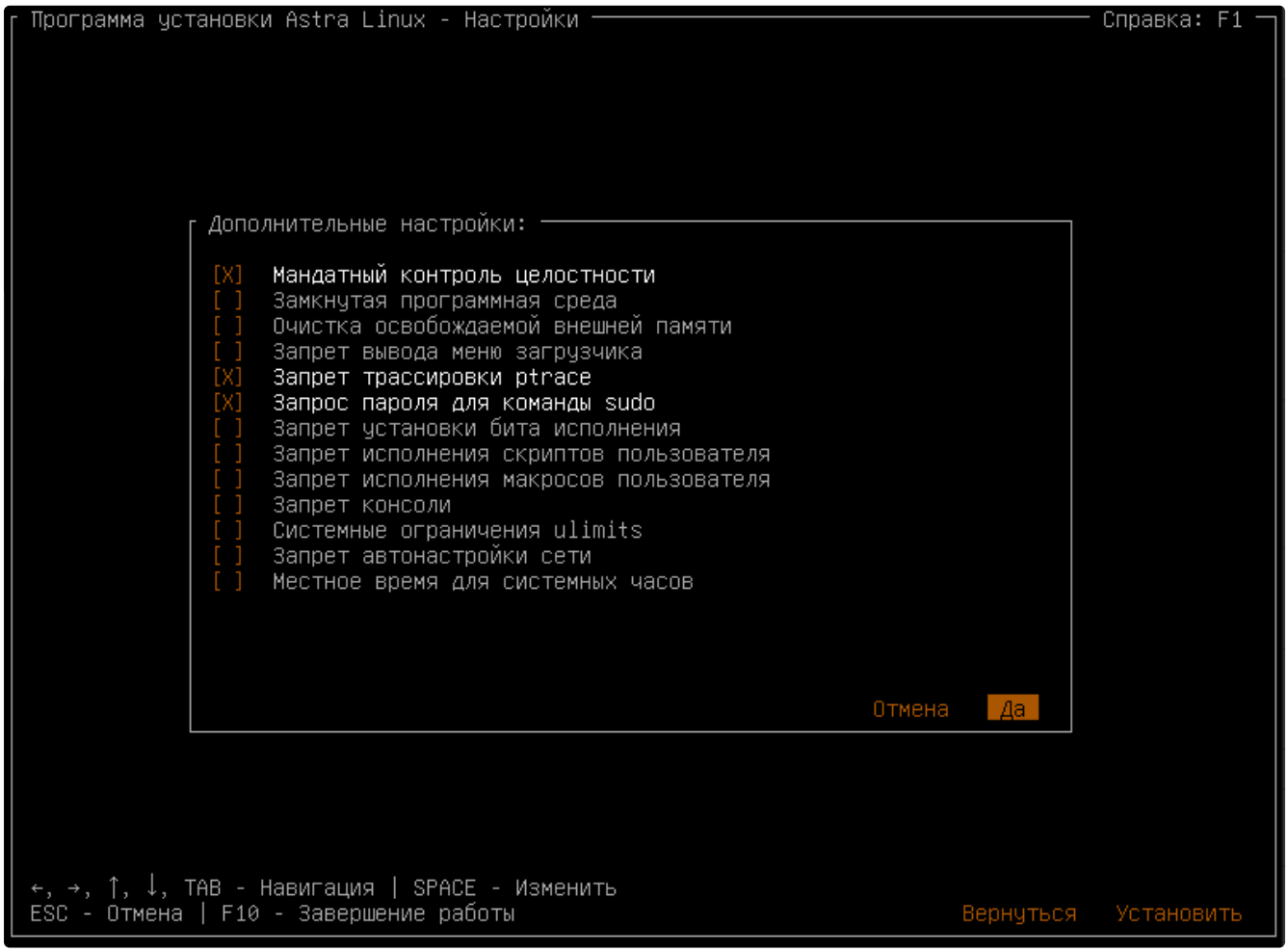
Согласится с Лицензионным Соглашением


Программа установки Astra Linux - Лицензия
Справка: F1
Лицензия:
Соглашения.
7.4 Пользователь соглашается на добровольную передачу своих персональных данных Компании, если предоставление соответствующих персональных данных необходимо для начала использования ПО. Пользователь выражает свое согласие на обработку (включая, но не ограничиваясь сбором и иным использованием) своих персональных данных Компании и/или его аффилированными лицами в соответствии с применимым законодательством и при условии соблюдения конфиденциальности и обеспечения защиты данных в соответствии с применимым законодательством. Все персональные данные будут обрабатываться исключительно в целях выполнения обязательств Компании по Соглашению или иных договоров, заключенных между Пользователем и Компанией в связи с использованием ПО. В случае если Пользователь предоставляет в том числе данные о юридическом лице или индивидуальном предпринимателе, Пользователь подтверждает, что имеет для этого достаточно правомочий, а предоставляемая Пользователем информация является полной и достоверной.
7.5 Компания вправе осуществлять контроль соблюдения Пользователем положений Соглашения и договоров, заключенных Пользователем с Компанией или Партнером Компании, путем запроса необходимой информации, анкетирования, а также получения доступа к вычислительной технике Пользователя, на которой осуществляется использование ПО на условиях и в объеме, предоставленного Компанией или Партнером Компании.
7.6 В отношении условий Соглашения и обязательств, вытекающих из Соглашения, применяется законодательство Российской Федерации.
Дата обновления документа: 27.11.2023
Выберите уровень защищенности, предусмотренный условиями лицензии:
( ) Базовый уровень защищенности («Орел»)
(*) Усиленный уровень защищенности («Воронеж»)
( ) Максимальный уровень защищенности («Смоленск»)
[X] Принимаю условия Лицензионного соглашения
←, →, ↑, ↓, TAB - Навигация | SPACE - Изменить
ESC - Отмена | F10 - Завершение работы
Далее

Настройки



Настойки (продолжение1)



Настройки (продолжение2)


Программа установки Astra Linux - Настройки
Справка: F1
Региональные настройки
Переключения языка: Ctrl+Shift
Системный язык: Русский
Часовой пояс: UTC+07:00 Россия (Новосибирск)
Дополнительные настройки: 27.11.2025, 09:12, NTP service d...
Настройки авторизации
Имя компьютера: astra-ecss
Имя администратора: eltex
Пароль: ***
Подтверждение пароля: ***
Настройки пароля GRUB: Совпадает с администраторским
Другие настройки
Компоненты ОС: Средства фильтрации сетевых паке...
Дополнительные настройки: Мандатный контроль целостности, ...
Дополнительные пакеты: Не выбраны
Ядро для установки: linux-6.1-generic (Рекомендуется)
Разметка диска: Использовать все пространство на...
[X] Перезагрузить компьютер после завершения установки
←, →, ↑, ↓, TAB - Навигация | SPACE - Изменить
ESC - Отмена | F10 - Завершение работы
Вернуться
Установить

Установка

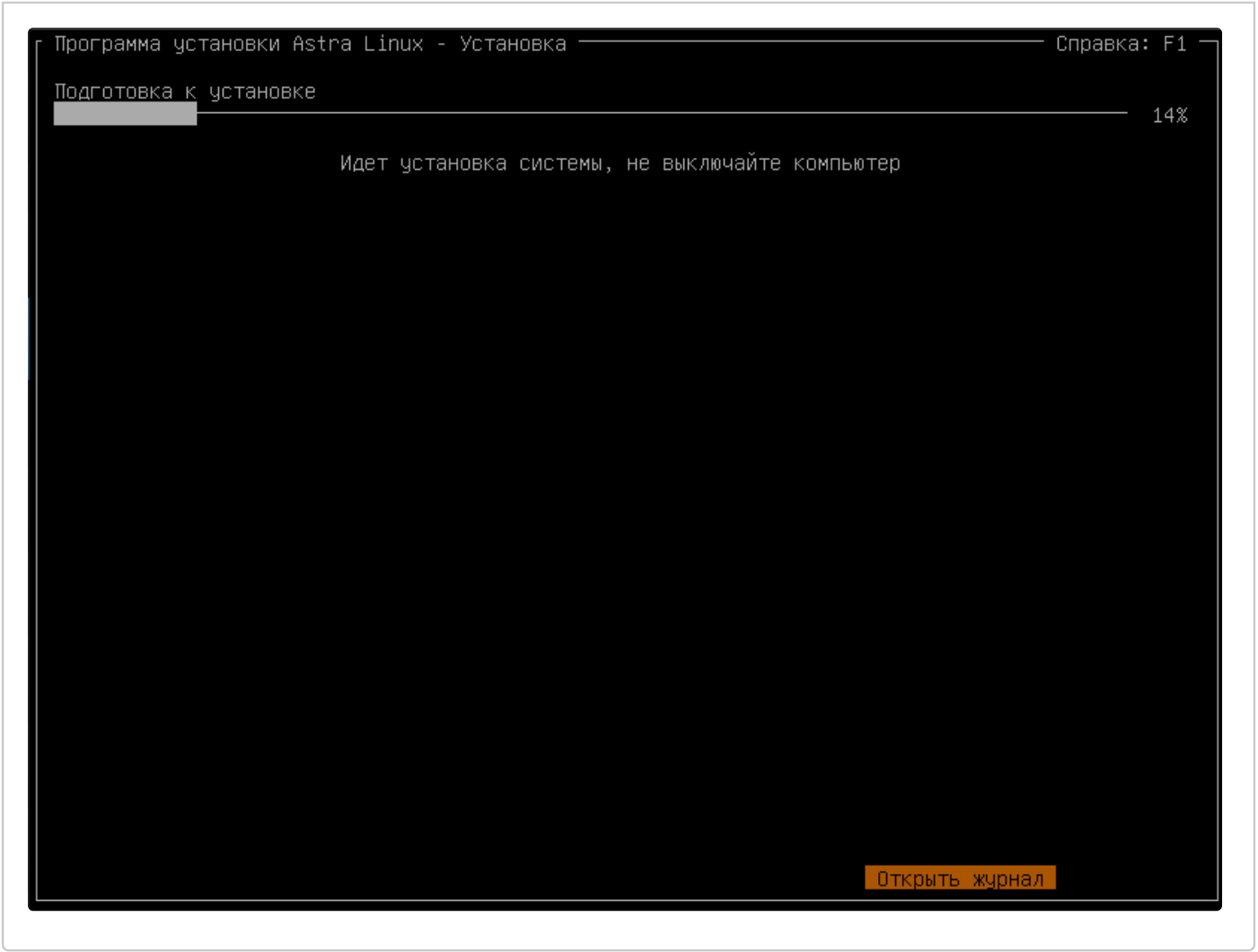


Таблица 1 — Вариант размещения информации в файловой системе на физических носителях для серверов

1	Загрузочный раздел операционной системы (создается автоматически)	boot	raid 1:hdd1,hdd2	boot	/boot	ext4	1 Gb	Первичный
2	Корневой раздел операционной системы	root	raid 1:hdd1,hdd2	root	/	ext4	30 Gb	Логический
3	Информация локальных баз данных	mnesia	raid 1:hdd1, hdd2	mnesia	/var/lib/ecss	ext4	30 Gb	Логический
4	Распределенная БД для хранения медиаресурсов	glusterfs	raid 1:hdd1, hdd2 или hdd3	glusterfs	/var/lib/ecss/glusterfs*	ext4	Max Gb	Логический

5	Журналы функционирования подсистем ОС	log	raid 1:hdd1,hdd2 или hdd3	log	/var/log	ext4	20 Gb	Логический
6	Журналы функционирования подсистем ECSS	ecss_log	raid 1:hdd1,hdd2 или hdd3	ecss_log	/var/log/ecss	ext4	20 Gb	Логический
7	Базы данных	ecss_db	raid 1:hdd1,hdd2 или hdd3	ecss_db	/srv/ecss/ecss-postgres-bdr-ssw/	ext4	100–400 Gb**	Логический
8	Файлы пользователя	home	raid 1:hdd1,hdd2 или hdd3	home	/home	ext4	10 Gb	Логический

⚠ * Если сервер не будет работать в кластере, то вместо **glusterfs** создается раздел **/var/lib/ecss/restfs**.
** Рекомендуемое значение для серий **Light, Light+, Midi** — 100 Gb. Рекомендуемое значение для серии **Heavy** — 200 Gb, **Super Heavy** — 400 Gb.

Для работы системы необходимо как **минимум** 256 Gb свободного пространства.

На серверах системы необходимо настроить параметр **"hostname".**

На всех серверах системы желательно указать одинаковое имя пользователя (любое, кроме **ssw**). Лицензия ECSS-10 привязывается к ключу eToken/ruToken и к имени компьютера (**hostname**). Системный пользователь **ssw** создается при инсталляции пакета **ecss-user**.

✔ При установке системы в кластере, **рекомендуемое** значение для первого сервера — **ecss1**, для второго — **ecss2**.

12.1.4 Проверка установки ОС

Проверка в основном сводится к правильности создания разделов диска и наличия доступа по ssh.

Для вывода информации о состоянии дискового пространства введите команду **df -h**. Она покажет общее и занятое место в разделах. Размеры разделов должны соответствовать проекту и введенным значениям при установке.

Для проверки доступа по ssh с машины, находящейся в одной подсети со вновь установленным сервером, нужно выполнить команду:

```
ssh <user>@<IP_ecss>
```

где:

- <user> — имя пользователя, заданного при установке;
- <IP_ecss> — IP-адрес хоста, заданного при установке.

12.1.5 Настройка /etc/hosts

Доменному имени хоста ecss1 должен соответствовать адрес 127.0.1.1. Также нужно прописать адрес хоста ecss2. Для этого в файле /etc/hosts необходимо прописать IP-адреса хостов ecss.

Например, для кластера: ecss1 имеет адрес 10.0.10.41, ecss2 — 10.0.10.42. Данные адреса нужно прописать в /etc/hosts:
посмотреть можно командой ***cat /etc/hosts***
изменить/добавить командой ***sudo nano /etc/hosts***
после модификации должен проходить пинг с сервера ecss1 ***ping ecss2*** и наоборот с сервера ecss2 ***ping ecss1***

на обоих хостах

```
sudo nano /etc/hosts
```

на ecss1

```
127.0.0.1 localhost
10.0.10.41 ecss1
10.0.10.42 ecss2
```

на ecss2

```
127.0.0.1 localhost
10.0.10.42 ecss2
10.0.10.41 ecss1
```

12.2 Конфигурация репозиториев

После установки для обновления будет использоваться только образ DVD диска, загрузка с репозитория отключена.

Давайте разрешим обновление пакетов с локального репозитория Eltex.

Копируем образ репозитория Eltex в локальную директорию.

на удаленном хосте

```
scp eltex_repo_latest.tar.gz eltex@10.0.20.41:/home/eltex
```

И распаковываем архив.

Пример:

на обоих хостах

```
tar -xzf eltex_repo_latest.tar.gz
```

Перемещаем директорию с Eltex пакетами в /var/spool/:

на обоих хостах

```
sudo mv eltex_repo /var/spool/
```

на обоих хостах

```
sudo apt-key add eltex_repo_gpg.key
```

Редактируем source.list

на обоих хостах

```
echo "deb file:/var/spool/eltex_repo 1.8_x86-64 contrib main non-free" | sudo tee -a /etc/apt/
sources.list
```

12.2.1 Изменение приоритета репозиториев

Для установки Erlang, Nodejs и Elixir из репозитория Eltex, необходимо изменить приоритет репозиториев для данных пакетов.

Добавить в файл /etc/apt/preferences.d/erlang строки:

на обоих хостах

```
sudo sh -c "echo -e 'Package: erlang*\nPin: version 2:24.3*\nPin-Priority: 1000' >> /etc/apt/
preferences.d/erlang"
```

Результат: cat /etc/apt/preferences.d/erlang

```
Package: erlang*
Pin: version 2:24.3*
Pin-Priority: 1000
```

Добавить в файл /etc/apt/preferences.d/nodejs строки:

на обоих хостах

```
sudo sh -c "echo -e 'Package: nodejs*\nPin: version 14.20.0-1nodesource1*\nPin-Priority: 1000'
>> /etc/apt/preferences.d/nodejs"
```

Результат: cat /etc/apt/preferences.d/nodejs

```
Package: nodejs*
Pin: version 14.20.0-1nodesource1*
Pin-Priority: 1000
```

Добавить в файл /etc/apt/preferences.d/elixir строки:

на обоих хостах

```
sudo sh -c "echo -e 'Package: elixir*\nPin: version 1.14*\nPin-Priority: 1000' >> /etc/apt/preferences.d/elixir"
```

Результат: cat /etc/apt/preferences.d/elixir

```
Package: elixir*
Pin: version 1.14*
Pin-Priority: 1000
```

Выполнить обновление:

на обоих хостах

```
sudo apt update
```

12.2.2 "Оптимизация" Операционной Системы

Выставить параметры ОС в режим производительности

Используем утилиту cpufrequtils.

на обоих хостах

```
sudo apt install cpufrequtils
```

по умолчанию после инсталляции ОС использует режим "ondemand" - "по запросу" (производительность CPU по запросу приложений, экономит электроэнергию , но ниже производительность):

на обоих хостах

```
cat /etc/init.d/cpufrequtils | grep GOVERNOR=
```

в выходном сообщении системы , режим работы по умолчанию после инсталляции - "ondemand" :

```
GOVERNOR="ondemand"
```

Установить режим - результативность/производительность - в файле /etc/init.d/cpufrequtils значение "**ondemand**" заменить на "**performance**"

на обоих хостах

```
sudo sed -i 's/GOVERNOR="ondemand"/GOVERNOR="performance"/g' /etc/init.d/cpufrequtils
```

Перезапустить утилиту:

на обоих хостах

```
sudo /etc/init.d/cpufrequtils restart
```

Затем выполнить команду:

на обоих хостах

```
sudo systemctl daemon-reload
```

Отключить SWAP

Ubuntu сервер SSW работает **в реальном масштабе времени** ,
поэтому **все** необходимые **данные должны находится в оперативной памяти,** использование файла подкачки (swap-файл - /swap. img) может привести к увеличению времени обработки вызовов приложения SSW ECSS10 , **что недопустимо**. Swap - отключаем.
Выполнить последовательно три команды:

Отключение Swap:

на обоих хостах

```
sudo swapoff -a
```

Удалить файл swap.img.

на обоих хостах

```
sudo rm /swapfile
```

Удалить эту строчку (//swapfile swap swap defaults 0 0)

на обоих хостах

```
sudo nano /etc/fstab
```

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
```

```
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda2 during curtin installation
UUID=4c88cb4a-42c9-4b52-98c2-78da7e23b621 /                       ext4    defaults
1 1
```

Для проверки выполните команду **free -h:**

на обоих хостах

```
free -h
```

Размер Swap равен 0 - то есть он выключен

```
free -h
              total        used        free      shared  buff/cache   available
Mem:           3,9G        110M        3,2G        820K        535M        3,5G
Swap:            0B          0B          0B
```

Установка часового пояса

При инсталляции astara linux **не предлагается** установить часовой пояс(по умолчанию устанавливается время UTC). Его нужно установить **вручную** (для корректной работы системы тарификации , работ по расписанию и т.д.), например:

на обоих хостах

```
sudo timedatectl set-timezone Asia/Novosibirsk
```

Улучшение работы высоконагруженных серверов

Улучшить работу высоконагруженных серверов можно увеличив лимит открытых файлов.

Для установки лимита открытых файлов необходимо:

Проверить текущий лимит командой:

на обоих хостах

```
ulimit -a
```

результат :

```
eltex@ecss1:~$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
scheduling priority             (-e) 0
file size               (blocks, -f) unlimited
pending signals                 (-i) 15515
max locked memory       (kbytes, -l) 65536
max memory size         (kbytes, -m) unlimited
open files                      (-n) 1024
pipe size            (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
real-time priority              (-r) 0
stack size              (kbytes, -s) 8192
cpu time               (seconds, -t) unlimited
max user processes              (-u) 15515
virtual memory          (kbytes, -v) unlimited
file locks                      (-x) unlimited
```

Данного лимита (open files **1024**) **недостаточно** для нормальной работы высоконагруженных серверов.

Установить лимит открытых файл для каждого пользователя:

на обоих хостах

```
sudo sed -i  '55i\*               soft    nproc           65536\n*               hard
nproc           131072\n*               soft    nofile          65536\n*               hard
nofile          131072\nroot            -       memlock         unlimited' /etc/security/
limits.conf
```

Результат:

```
# /etc/security/limits.conf
#
#Each line describes a limit for a user in the form:
#
#<domain>        <type>  <item>  <value>
#
#Where:
#<domain> can be:
#        - a user name
#        - a group name, with @group syntax
#        - the wildcard *, for default entry
#        - the wildcard %, can be also used with %group syntax,
#                 for maxlogin limit
#        - NOTE: group and wildcard limits are not applied to root.
#          To apply a limit to the root user, <domain> must be
```

```
#          the literal username root.
#
#<type> can have the two values:
#        - "soft" for enforcing the soft limits
#        - "hard" for enforcing hard limits
#
#<item> can be one of the following:
#        - core - limits the core file size (KB)
#        - data - max data size (KB)
#        - fsize - maximum filesize (KB)
#        - memlock - max locked-in-memory address space (KB)
#        - nofile - max number of open files
#        - rss - max resident set size (KB)
#        - stack - max stack size (KB)
#        - cpu - max CPU time (MIN)
#        - nproc - max number of processes
#        - as - address space limit (KB)
#        - maxlogins - max number of logins for this user
#        - maxsyslogins - max number of logins on the system
#        - priority - the priority to run user process with
#        - locks - max number of file locks the user can hold
#        - sigpending - max number of pending signals
#        - msgqueue - max memory used by POSIX message queues (bytes)
#        - nice - max nice priority allowed to raise to values: [-20, 19]
#        - rtprio - max realtime priority
#        - chroot - change root to directory (Debian-specific)
#
#<domain>      <type>  <item>         <value>
#

#*               soft    core            0
#root            hard    core            100000
#*               hard    rss             10000
#@student        hard    nproc           20
#@faculty        soft    nproc           20
#@faculty        hard    nproc           50
#ftp             hard    nproc           0
#ftp             -       chroot          /ftp
#@student        -       maxlogins       4
```

```
*               soft    nproc           65536
*               hard    nproc           131072
*               soft    nofile          65536
*               hard    nofile          131072
root            -       memlock         unlimited

# End of file
```

12.3 Настройка сетевых интерфейсов

⚠ Получение адресов на сетевых интерфейсах по DHCP **недопустимо** на серверах ECSS.

Сетевые настройки необходимо выполнять с помощью Netplan.

По умолчанию в Astra Linux пакет netplan не установлен.

Установить пакеты для обслуживание функции netplan:

на обоих хостах

```
sudo apt install -y dialog libssl1.0.0 netplan.io
```

Сейчас можно выполнить конфигурацию netplan:

на обоих хостах

```
sudo nano /etc/netplan/ecss.yaml
```

12.3.1 Пример netplan для режима active-backup

Netplan для интерфейсов сервера ecss1 (/etc/netplan/ecss.yaml)

на ecss1

```
# Netplan для хоста ecss1 программного
коммутатора
# Обратите внимание на обязательное наличие
минимум двух пробелов в каждой строке и
секции (кроме строки секции network)

network:
  version: 2 # Версия netplan
  renderer: networkd # Исполнитель
конфигураций netplan
  ethernets: # Раздел описания интерфейсов
 ethernet
    enp0s3: # Название внутреннего
интерфейса ВМ
      dhcp4: no # Отключаем на интерфейсах
динамическое распределение IP-адреса
      addresses: [192.168.56.41/24] #
Интерфейс управления OAM
      gateway4: 192.168.56.1
    enp0s8: # Название внешнего интерфейса
ВМ
      dhcp4: no # Отключаем на интерфейсах
динамическое распределение IP-адреса
  vlans:
    net.10: # Интерфейс управления/Postgres
        id: 10
        link: enp0s8
        addresses: [10.0.10.41/24]# Адрес
для Postgres
    net.20: # Интерфейс для VoIP
      id: 20
      link: enp0s8
      addresses: [10.0.20.41/24]# Адрес для
 VoIP / SIP
      routes:
        - to: 10.0.20.0/24
          via: 10.0.20.1
```

Netplan для интерфейсов сервера ecss2 (/etc/netplan/ecss.yaml)

на ecss2

```
# Netplan для хоста ecss2 программного
коммутатора
# Обратите внимание на обязательное наличие
минимум двух пробелов в каждой строке и
секции (кроме строки секции network)

network:
  version: 2 # Версия netplan
  renderer: networkd # Исполнитель
конфигураций netplan
  ethernets: # Раздел описания интерфейсов
 ethernet
    enp0s3: # Название внутреннего
интерфейса ВМ
      dhcp4: no # Отключаем на интерфейсах
динамическое распределение IP-адреса
      addresses: [192.168.56.42/24] #
Интерфейс управления OAM
      gateway4: 192.168.56.1
    enp0s8: # Название внешнего интерфейса
ВМ
      dhcp4: no # Отключаем на интерфейсах
динамическое распределение IP-адреса
  vlans:
    net.10: # Интерфейс управления/Postgres
        id: 10
        link: enp0s8
        addresses: [10.0.10.42/24]# Адрес
для Postgres
    net.20: # Интерфейс для VoIP
      id: 20
      link: enp0s8
      addresses: [10.0.20.42/24]# Адрес для
 VoIP / SIP
      routes:
        - to: 10.0.20.0/24
          via: 10.0.20.1
```

Так же проверить отсутствие в каталоге **/etc/netplan/** других файлов, если другие файлы присутствуют , то их нужно переместить в другой каталог или удалить, в противном случае возможна некорректная настройка сетевых интерфейсов и некорректная работа SSW.

на обоих хостах

```
ls -la /etc/netplan/
```

Применить, установленные параметры командой:

на обоих хостах

```
sudo netplan apply
```

Просмотреть получившиеся настройки можно при помощи команд **ifconfig** или **ip a**:

Для применения новых сетевых настроек необходимо выполнить команду **netplan apply.** Перезапуск сети или системы не требуется. Подробнее про настройки netplan см. в Приложении E. Netplan.

12.4 Обновление ОС и инсталляция необходимого ПО

12.4.1 Обновление системы

Перед началом установки необходимо обновить ОС:

на обоих хостах

```
sudo apt update
```

на обоих хостах

```
sudo apt upgrade --enable-upgrade
```

Доступ между серверами без использования пароля

Далее следует разрешить доступ между серверами с через ssh по rsa-ключам - **без использования пароля**.

На хосте *ecss1 (затем на ecss2)* сгенерировать rsa-ключ следующий командой (выполнить **без sudo**, чтобы ключ был создан для текущего пользователя), на вопросы, использовать значение **по умолчанию**, нажав **три раза Enter**:

на обоих хостах

```
ssh-keygen
```

Скопировать rsa-ключ на сервер партнера ecss2 командой - **ssh-copy-id** ecss2.

на ecss1

```
ssh-copy-id ecss2
```

Скопировать rsa-ключ на сервер партнера ecss1 командой - **ssh-copy-id** ecss1.

на ecss2

```
ssh-copy-id ecss1
```

Проверить подключение между серверами, со стороны сервера ecss1 - **ssh ecss2** соединение должно быть установлено **без запроса пароля** . По окончанию теста вернутся обратно на сервер ecss1 , командой **exit.**

на ecss1

```
ssh ecss2
```

Аналогичным образом со стороны сервера ecss2 - **ssh ecss1** соединение должно быть установлено **без запроса пароля .** По окончанию теста вернутся обратно на сервер ecss2 , командой **exit**.

на ecss2

```
ssh ecss1
```

12.4.2 Инсталляция необходимого ПО

Установка дополнительных пакетов программ

Для работы SSW, потребуются дополнительные пакеты ПО не установленные в Astra Linux по умолчанию, но доступные в "eltex_repo_latest" репозитории.

выполнить сначала на ecss1, затем на ecss2

```
sudo apt install ntp iptables tcpdump vlan dnsmasq aptitude atop ethtool htop mc minicom mtr-tiny nmap pptpd  screen ssh tftpd-hpa sngrep tshark  gnuplot debconf-utils keepalived attr bridge-utils ncdu wget
```

Пакеты ПО которые рекомендуется установить для работы

Список **обязательного** сервисного программного обеспечения:

```
sudo apt install ntp tcpdump vlan dnsmasq
```

ntp	NTP-сервер
tcpdump	сниффер пакетов
vlan	управление VLAN
dnsmasq	легковесный DNS/DHCP-сервер

Список **рекомендуемого** диагностического и вспомогательного программного обеспечения:

```
sudo apt install aptitude atop ethtool htop mc screen ssh tftpd sngrep tshark gnuplot
debconf-utils iotop ncdu
```

aptitude	установка программ из репозиториев, рекомендуется использовать вместо программы apt/apt-get
atop	мониторинг загрузки хоста с функцией периодического сохранения информации в файлы
ethtool	просмотр статистики сетевых интерфейсов
htop	мониторинг процессов
mc	файловый менеджер
screen	мультиплексор терминалов
ssh	сервер и клиент SSH
tftpd	TFTP-сервер
sngrep	трассировка sip
tshark	консольный аналог wireshark
gnuplot	вывод графиков статистики
debconf-utils	набор утилит для работы с базой debconf
iotop	инструмент для мониторинга использования ввода-вывода (IO) на диске в реальном времени в Linux
ncdu	утилита для поиска больших директорий в системе Linux

⚠ Данное программное обеспечение не требуется для работы системы ECSS-10, однако может упростить сервисное обслуживание системы и её отдельных компонентов со стороны инженеров эксплуатации и техподдержки.

Список **обязательных** пакетов для схем **с резервированием**:

```
sudo apt install ifenslave-2.6 keepalived attr
```

ifenslave-2.6	управление BOND-интерфейсами
keepalived	сервис мониторинга серверов/служб в кластере
attr	сервис управления атрибутами файловых систем

Список **дополнительных** пакетов для схем **с резервированием**:

```
sudo apt install bridge-utils ethtool
```

bridge-utils	управление bridge-интерфейсами
ethtool	управление и мониторинг сетевых интерфейсов

Для **просмотра** установленных пакетов выполнить следующую команду (**данный пункт является необязательным**: его можно выполнить, если не уверены, что какие-то пакеты установлены):

на обоих хостах

```
sudo dpkg --get-selections
```

⚠ Перед началом установки пакетов ecss , нужно убедится в соответствии полосы пропускания СПД необходимым требованиям.
Для этого выполнить команду **sudo ethtool <имя интерфейса>** для всех физических интерфейсов.
Пример:

```
sudo ethtool net.20
Settings for net.20:
    Supported ports: [ TP ]
    Supported link modes:   10baseT/Half 10baseT/Full
                            100baseT/Half 100baseT/Full
                            1000baseT/Full
    Supported pause frame use: No
    Supports auto-negotiation: Yes
    Supported FEC modes: Not reported
    Advertised link modes:  10baseT/Half 10baseT/Full
                            100baseT/Half 100baseT/Full
                            1000baseT/Full
    Advertised pause frame use: No
    Advertised auto-negotiation: Yes
    Advertised FEC modes: Not reported
    Speed: 1000Mb/s
    Duplex: Full
    Auto-negotiation: on
    Port: Twisted Pair
    PHYAD: 0
    Transceiver: internal
    MDI-X: off (auto)
    Link detected: yes
```

Проверить значение следующих параметров:
Advertised auto-negotiation: **Yes**
Speed: **1000Mb/s** (не менее)
Duplex: **Full**

12.5 Инсталляция пакетов ECSS

12.5.1 Предварительные требования

- Установленная и обновленная операционная система Astra Linux;
- Отсутствие в системе пользователя с именем **ssw**;
- Разбиение дискового пространства в соответствии с рекомендациями;
- Настроенная сеть;
- Установленный набор необходимых пакетов;

- Наличие локального файла репозитория Eltex.

установка пакетов deb

```
   ┌ версия ПО ECSS. Сейчас в заявке на сертификацию она заявлена как 3.

   │ Мы её менять не будем до следующей сертификации или
  │ выпуска принципиально новой версии системы.

   │    ┌ версия System Release. Общая версия для всех компонент входящих в конкретный релиз.

   │       │ Меняем её централизовано,
   │   │ принимаем решение о выпуске нового релиза. Релизы между собой обычно не совместимы.
┌──┘ ┌──┘
ECSS.SysRel.SubMaj.SubMin
              └────┘ └────┘
        │    └ версия Subsystem minor. Минорную версию подсистемы устанавливает разработчик
подсистемы.

                  │          Минорную версию
        │     меняем при добавлении патча. Как правило, минорные версии в рамках одной мажорной
версии
        │     совместимы между собой и отличаются в рамках конкретных патчей.

                  └ версия Subsystem major. Мажорная версия устанавливается разработчиком
подсистемы.

                     Мажорную версию подсистемы необходимо менять при внесении в подсистему
существенных изменений.
```

Утилита установки пакетов APT анализирует версию пакета слева направо, для примера, мы имеем пакет 14.14.7.7 ,
в репозитории находятся пакеты
14.14.7.8
14.14.7.9
14.14.8.1
14.14.20 -14.14.28
то при выполнении команды **sudo apt install имя пакета** → будет автоматически проверен и установлен пакет 14.14.28 , так как он самый последний, анализ будет выполнен по 3-й позиции (major), анализ по 4-й (minor) выполнен не будет (аналогичным образом будет выполнятся команда **sudo apt upgrade --enable-upgrade**).
В случае если для конкретной ситуации требуется перейти с версии 14.14.7.7 на версию 14.14.7.9 , стандартная команда → **sudo apt upgrade --enable-upgrade** , нам не поможет, так как будет выбран самый новый пакет, в данной ситуации нам **необходимо в явном виде** указать какую версию пакета мы хотим установить, в данном примере мы **должны** выбрать команду → **sudo apt install имя пакета=14.14.7.9** . Обычно это необходимо для тестирования определенного патча, для стандартных обновлений достаточно выбора привычной команды установки/обновления пакета

В ходе установки пакетов ECSS нужно будет ответить на ряд вопросов для формирования необходимой конфигурации. Для автоматической загрузки требуемых настроек можно воспользоваться командами из пакета debconf-utils.
Описание и примеры использования при работе с debconf приведены в Приложении B. Debconf.

Для инсталляции системы ECSS-10 необходимо устанавливать пакеты в порядке, в котором они описаны ниже в документации.

12.5.2 Инсталляция обязательных пакетов

Установка пакета ecss-dns-env

Выполнить следующую команду:

на обоих хостах

```
sudo apt install -y ecss-dns-env
```

Настройщик предложит выбрать разделы для настройки по вопросам ниже. **Нужно выбрать broker.** (для примера 10.0.10.51 и 10.0.10.52)

Вопрос ecss-dns-env	Ответы для ecss1	Ответы для ecss2	Пример
[Primary broker] Введите адрес ([Primary broker] Enter ip)	**10.0.10.41** (необходимо ввести)	**10.0.10.41** (необходимо ввести)	
[Secondary broker] Введите адрес ([Secondary broker] Enter ip)	**10.0.10.42** (необходимо ввести)	**10.0.10.42** (необходимо ввести)	

ВАЖНО

В случае последующей коррекции IP-адресов следует использовать команду:

```
sudo dpkg-reconfigure ecss-dns-env
```

Настройка преобразование DNS имени в IP адрес

Для корректной работы системы, после установки пакета ecss-node требуется скорректировать дефолтные установки . Для этого:

В файле /etc/resolv.conf изменить параметр nameserver установленный системой по умолчанию на 127.0.0.1 (если строк nameserver несколько - остальные удалить/закомментировать):

```
sudo nano /etc/resolv.conf
```

Пример:

```
domain eltex.loc
search eltex.loc
nameserver 127.0.0.1
```

В Astra Linux после перезагрузки файл /etc/resolv.conf приводится к исходному виду, поэтому его **нужно сделать неизменяемым**:

```
sudo chattr +i /etc/resolv.conf
```

Для доступа к репозиториям AstraLinux и Элтекс нужно указать локальный DNS или глобальный (для примера 8.8.8.8 или 77.88.8.8)

Добавить в файл /etc/dnsmasq.d/ecss строки server=<Ipadd DNS сервера> (в примере 77.88.8.8) :

```
sudo sh -c "echo 'server=77.88.8.8' >> /etc/dnsmasq.d/ecss"
```

Перезапустить сервис:

```
sudo systemctl restart dnsmasq.service
```

Установка Postgres

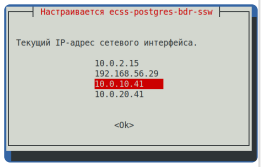
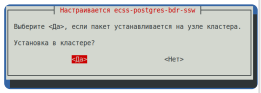
Установка пакета ecss-postgres-bdr-ssw

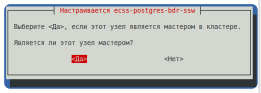
Выполняется последовательно, первым устанавливается на сервере который работает в режиме "Master"

ecss1

```
sudo apt install -y ecss-postgres-bdr-ssw
```

При конфигурации пакета **ecss-postgres-bdr-ssw** будут заданы вопросы:

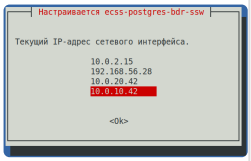
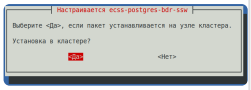
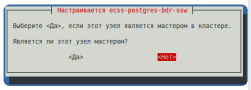
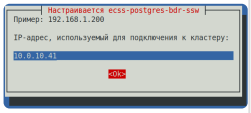
Вопросы ecss-postgres-bdr-ssw	Ответы для ecss1 (Master)	Пример
Текущий IP-адрес сетевого интерфейса.	**Выбрать** из списка IP-адрес для работы с Postgres	
Установка в кластере?	Yes (значение по умолчанию)	

Вопросы ecss-postgres-bdr-ssw	Ответы для ecss1 (Master)	Пример
Является ли этот узел мастером?	**Yes** (необходимо выбрать)	

ecss2

```
sudo apt install -y ecss-postgres-bdr-ssw
```

При конфигурации пакета **ecss-postgres-bdr-ssw** будут заданы вопросы:

Вопросы ecss-postgres-bdr-ssw	Ответы для ecss2 (Slave)	Пример
Текущий IP-адрес сетевого интерфейса.	**Выбрать** из списка IP-адрес для работы с Postgres	
Установка в кластере?	Yes (значение по умолчанию)	
Является ли этот узел мастером?	No (значение по умолчанию)	
IP-адрес, используемый для подключения к кластеру:	**Указать** IP-адрес Master (ecss1)	

Проверить корректность настройки репликации

Во время установки в директорию `/srv/ecss/ecss-postgres-bdr-ssw` устанавливаются скрипт **postgresbdr_ssw.sh**

Для проверки репликации выполнить следующие команды:

Для проверки репликации выполнить команду на обоих хостах:

```
cd /srv/ecss/ecss-postgres-bdr-ssw
```

Для проверки репликации выполнить команду на обоих хостах:

```
sudo ./postgresbdr_ssw.sh check
```

Вывод на хостах **должен быль одинаковый:**

ecss1

```
sudo ./postgresbdr_ssw.sh check
Enter master ip please
Like 10.150.150.15
10.0.10.41
Enter slave ip please
10.0.10.42
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
таблица нод на слейве по ecss_storekeeper_db
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+------------------------------------------------------------------------------
----------
 7577617967861174292 | host=10.0.10.41 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7577618825220177940 | host=10.0.10.42 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 строки)

таблица коннекций на слейве по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+------------------------------------------------------------------------------
----------
 7577617967861174292 | host=10.0.10.41 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
```

```
 7577618825220177940 | host=10.0.10.42 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 строки)
```

ecss2

```
sudo ./postgresbdr_ssw.sh check
Enter master ip please
Like 10.150.150.15
10.0.10.41
Enter slave ip please
10.0.10.42
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
таблица нод на слейве по ecss_storekeeper_db
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+------------------------------------------------------------------------------
----------
 7577617967861174292 | host=10.0.10.41 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7577618825220177940 | host=10.0.10.42 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 строки)

таблица коннекций на слейве по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+------------------------------------------------------------------------------
----------
 7577617967861174292 | host=10.0.10.41 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
```

```
 7577618825220177940 | host=10.0.10.42 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 строки)
```

пример с ошибками

нет доступа до slave

```
sudo ./postgresbdr_ssw.sh check
[sudo] password for abf:
Sorry, try again.
[sudo] password for abf:
Enter master ip please
Like 10.150.150.15
10.0.10.51
Enter slave ip please
10.0.10.52
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
     node_sysid      |                                    node_local_dsn
                                  | node_init_from_dsn
---------------------
+-------------------------------------------------------------------------
-------------+--------------------
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1 |
(1 row)
таблица нод на слейве по ecss_storekeeper_db
psql: error: connection to server at "10.0.10.52", port 5439 failed:
Connection refused
    Is the server running on that host and accepting TCP/IP connections?
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                        conn_dsn
                                  |
---------------------
+-------------------------------------------------------------------------
```

```
-------------
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(1 row)
таблица коннекций на слейве по ecss_storekeeper_db
psql: error: connection to server at "10.0.10.52", port 5439 failed:
Connection refused
    Is the server running on that host and accepting TCP/IP connections?
```

ошибка несоответствия коннекций

```
sudo ./postgresbdr_ssw.sh check
Enter master ip please
Like 10.150.150.15
10.0.10.51
Enter slave ip please
10.0.10.52
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q – выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
     node_sysid      |                                        node_local_dsn
                                  | node_init_from_dsn
---------------------
+-------------------------------------------------------------------------
-------------+--------------------
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1 |
(1 row)
таблица нод на слейве по ecss_storekeeper_db
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                          conn_dsn
                                 |
---------------------
+-------------------------------------------------------------------------
-------------
```

```
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(1 row)

таблица коннекций на слейве по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

--------------------
+---------------------------------------------------------------------------
-------------
 7569499877813944343 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.52 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(3 rows)
```

Установка ecss-node

Установка обязательного пакета **ecss-node** включает в себя установку и первоначальную настройку основных подсистем.

В системе уже должен быть установлен пакет **ecss-postgres-bdr-ssw**

Для установки пакета **ecss-node** выполните команду:

первоначально на ecss1

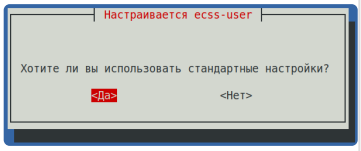
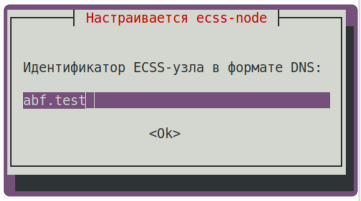
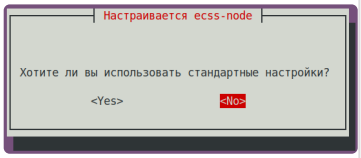
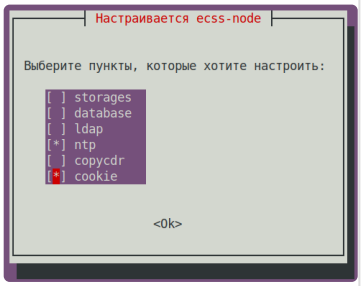
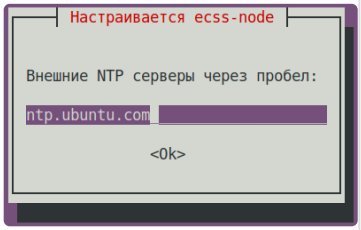
```
sudo apt install -y ecss-node
```

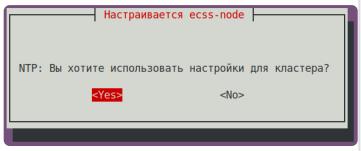
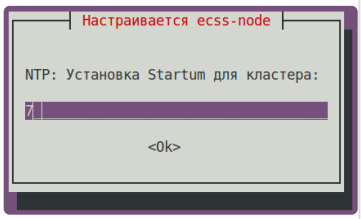
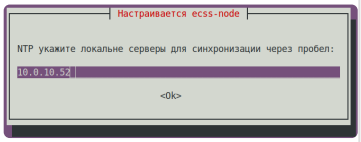
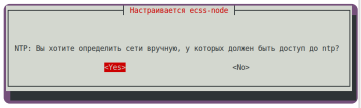
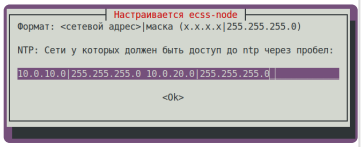
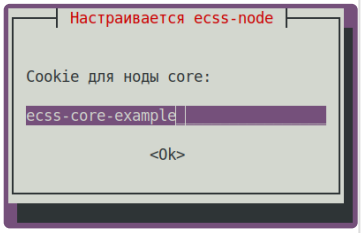
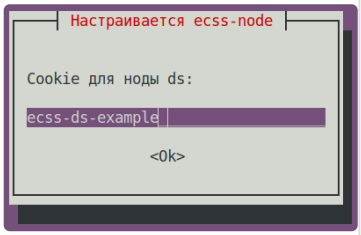
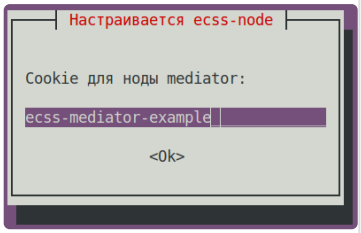
Во время установки пакета создается пользователь **ssw**, от имени которого запускаются все сервисы ecss*. Создаются необходимые каталоги, выполняется настройка DNS, идет настройка SSL-сертификатов.

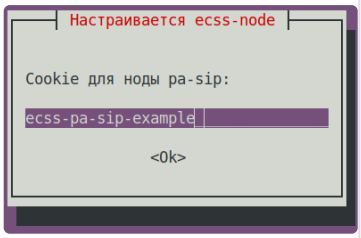
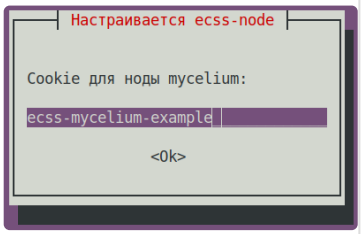
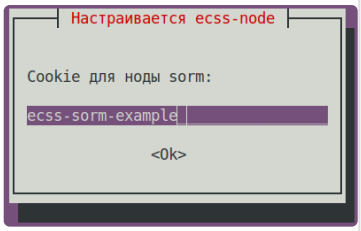
При установки так же будет установлен пакет **ecss-user**

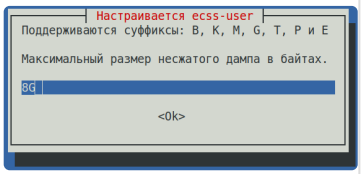
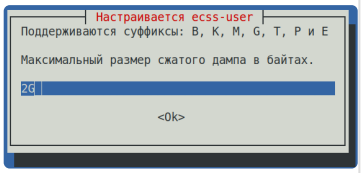
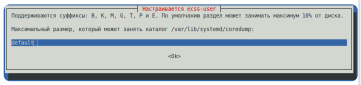
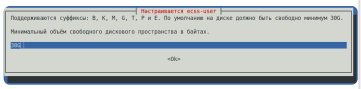
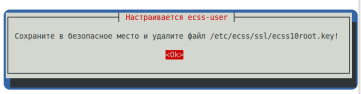
Установка на сервере **ecss1**

Во время установки будет предложено настроить параметры, необходимые для формирования конфигурационных файлов, примеры ответов следует ниже.

Вопросы ecss-user	Ответы для ecss1	Пример
Хотите ли вы использовать стандартные настройки?	Yes (значение по умолчанию)	
Вопросы ecss-node	**Ответы для ecss1**	**Пример**
Идентификатор ECSS-узла в формате DNS (Настройка имени кластера): (любое уникальное имя, в примере abf.test)	**abf.test** (необходимо ввести) Формат **ECSS_ID** Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире. `[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]`	
Хотите ли вы использовать стандартные настройки?	**No** (необходимо выбрать)	
Выберите пункты, которые хотите настроить:	**ntp + cookie** (необходимо выбрать)	
Внешние NTP серверы через пробел:	ntp.ubuntu.com (значение по умолчанию)	

NTP: Вы хотите использовать настройки для кластера?	Yes (значение по умолчанию)	
NTP: Установка Startum для кластера:	7 (значение по умолчанию)	
NTP укажите локальне серверы для синхронизации через пробел: (в примере ecss2 - 10.0.10.42)	**10.0.10.42** (необходимо ввести)	
NTP: Вы хотите определить сети вручную, у которых должен быть доступ до ntp?	Yes (значение по умолчанию)	
NTP: Сети у которых должен быть доступ до ntp через пробел: (в примере 10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0)	**10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0** (необходимо ввести)	
Cookie для ноды core: (указать уникальный cookie для core , в примере ecss-core-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-core-example** (необходимо ввести)	
Cookie для ноды ds: (указать уникальный cookie для ds , в примере ecss-ds-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-ds-example** (необходимо ввести)	
Cookie для ноды mediator: (указать уникальный cookie для mediator , в примере ecss-mediator-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mediator-example** (необходимо ввести)	

Cookie для ноды pa-sip: (указать уникальный cookie для pa-sip , в примере ecss-pa-sip-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-pa-sip-example** (необходимо ввести)	
Cookie для ноды mycelium: (указать уникальный cookie для mycelium , в примере ecss-mycelium-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mycelium-example** (необходимо ввести)	
Cookie для ноды sorm: (указать уникальный cookie для sorm , в примере ecss-sorm-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-sorm-example** (необходимо ввести)	

Вопросы ecss-user	**Ответы для ecss1**	**Пример**
Максимальный размер несжатого дампа в байтах.	8G (значение по умолчанию)	
Максимальный размер сжатого дампа в байтах.	2G (значение по умолчанию)	
Максимальный размер, который может занять каталог /var/lib/systemd/coredump:	default (значение по умолчанию)	
Минимальный объем свободного дискового пространства в байтах.	30G (значение по умолчанию)	
Сохраните в безопасное место и удалите файл /etc/ecss/ssl/ecss10root.key!	Ok (значение по умолчанию)	

После установки пакета ecss-node на хосте ecss1, выполните команду **"sudo usermod -a -G ssw <Имя_пользователя>" на хосте ecss1**.

Пример:

```
sudo usermod -a -G ssw eltex
```

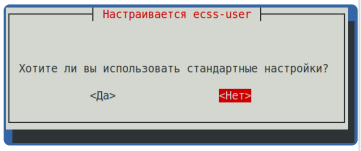
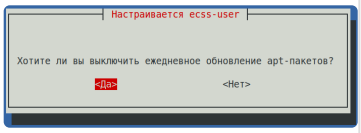
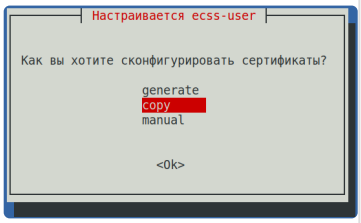
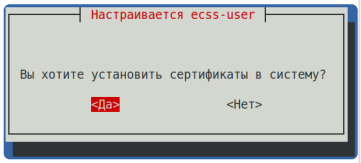
Затем **выполите установку на хосте ecss2.**
Используйте режим "Скопировать существующие сертификаты (copy) по ssh" с сервера ecss1

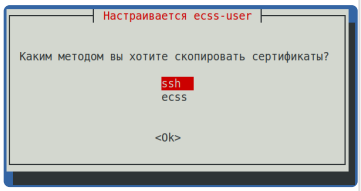
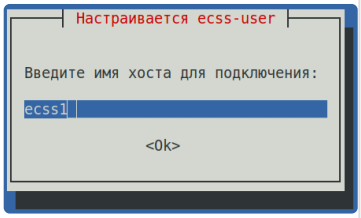
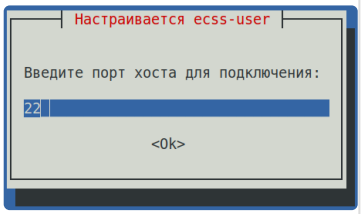
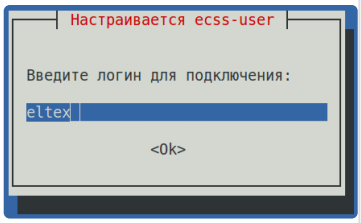
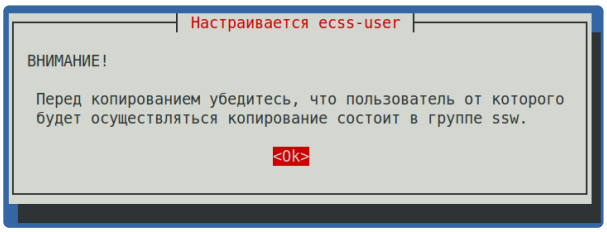
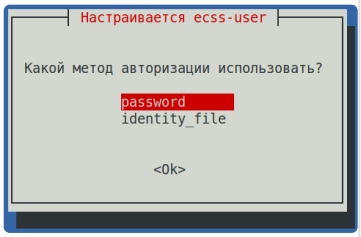
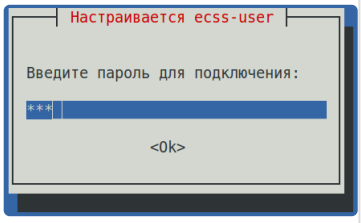
Установка на сервере **ecss2**

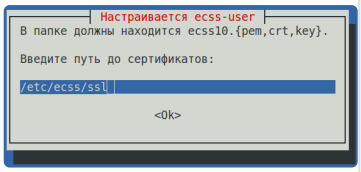
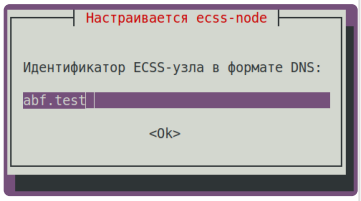
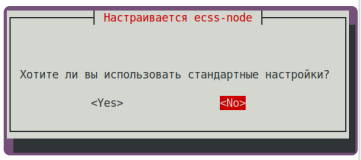
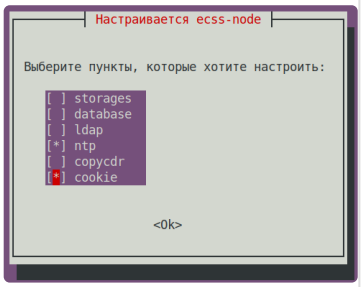
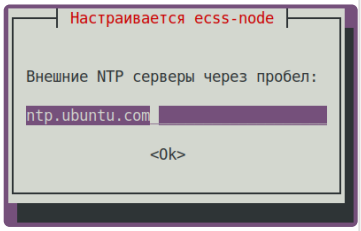
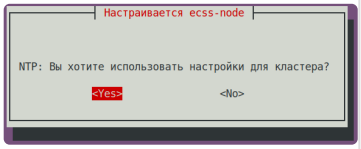
на ecss2

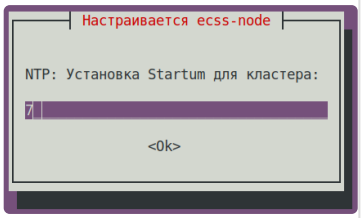
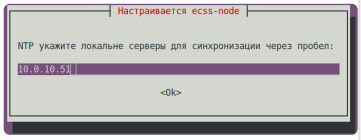
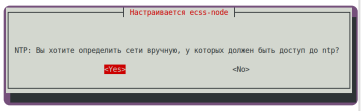
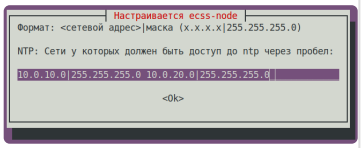
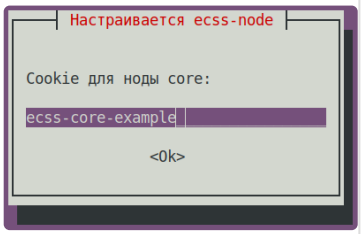
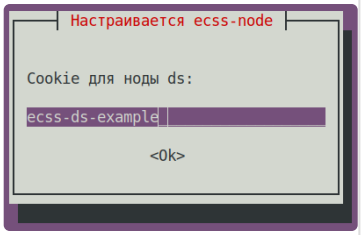
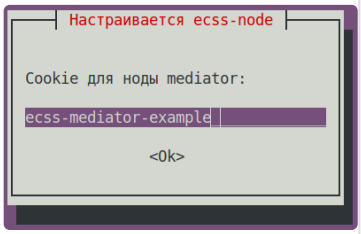
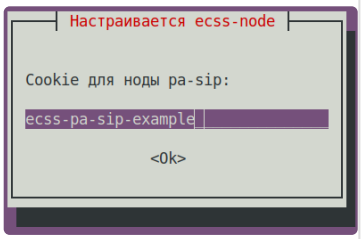
```
sudo apt install -y ecss-node
```

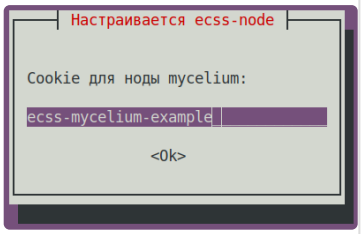
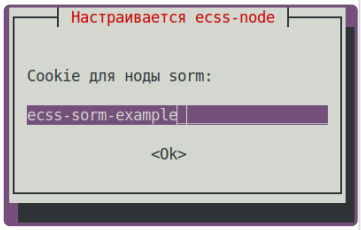
Во время установки будет предложено настроить параметры, необходимые для формирования конфигурационных файлов, примеры ответов следует ниже.

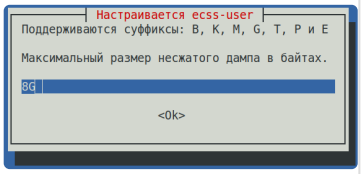
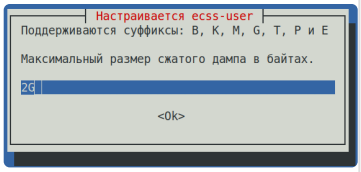
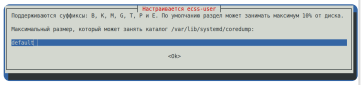
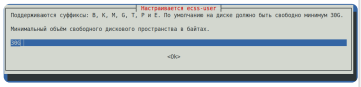
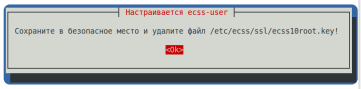
Вопросы ecss-user	Ответы для ecss2	Пример
Хотите ли вы использовать стандартные настройки?	No (значение по умолчанию)	
Хотите ли вы выключить ежедневное обновление apt-пакетов?	Yes (значение по умолчанию)	
Как вы хотите сконфигурировать сертификаты?	**copy** (необходимо выбрать)	
Вы хотите установить сертификаты в систему?	Yes (значение по умолчанию)	

Каким методом вы хотите скопировать сертификаты?	**ssh** (необходимо выбрать)	
Введите имя хоста для подключения:	**ecss1** (необходимо ввести)	
Введите порт хоста для подключения:	22 (значение по умолчанию)	
Введите логин для подключения: (для примера eltex)	**eltex** (необходимо ввести)	
		
Какой метод авторизации использовать?	**password** (необходимо выбрать)	
Введите пароль для подключения:		

Введите путь до сертификатов:	/etc/ecss/ssl (значение по умолчанию)	
Вопросы ecss-node	**Ответы для ecss2**	**Пример**
Идентификатор ECSS-узла в формате DNS (Настройка имени кластера): (любое уникальное имя, в примере abf.test)	**abf.test** (необходимо ввести) Формат **ECSS_ID** Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире. `[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]`	
Хотите ли вы использовать стандартные настройки?	**No** (необходимо выбрать)	
Выберите пункты, которые хотите настроить:	**ntp + cookie** (необходимо выбрать)	
Внешние NTP серверы через пробел:	ntp.ubuntu.com (значение по умолчанию)	
NTP: Вы хотите использовать настройки для кластера?	Yes (значение по умолчанию)	

NTP: Установка Startum для кластера:	7 (значение по умолчанию)	
NTP укажите локальне серверы для синхронизации через пробел: (в примере ecss2 - 10.0.10.41)	**10.0.10.41** (необходимо ввести)	
NTP: Вы хотите определить сети вручную, у которых должен быть доступ до ntp?	Yes (значение по умолчанию)	
NTP: Сети у которых должен быть доступ до ntp через пробел: (в примере 10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0)	**10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0** (необходимо ввести)	
Cookie для ноды core: (указать уникальный cookie для core , в примере ecss-core-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-core-example** (необходимо ввести)	
Cookie для ноды ds: (указать уникальный cookie для ds , в примере ecss-ds-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-ds-example** (необходимо ввести)	
Cookie для ноды mediator: (указать уникальный cookie для mediator , в примере ecss-mediator-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mediator-example** (необходимо ввести)	
Cookie для ноды pa-sip: (указать уникальный cookie для pa-sip , в примере ecss-pa-sip-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-pa-sip-example** (необходимо ввести)	

Cookie для ноды mycelium: (указать уникальный cookie для mycelium , в примере ecss-mycelium-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mycelium-example** (необходимо ввести)	
Cookie для ноды sorm: (указать уникальный cookie для sorm , в примере ecss-sorm-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-sorm-example** (необходимо ввести)	

Вопросы ecss-user	**Ответы для ecss2**	**Пример**
Максимальный размер несжатого дампа в байтах.	8G (значение по умолчанию)	
Максимальный размер сжатого дампа в байтах.	2G (значение по умолчанию)	
Максимальный размер, который может занять каталог /var/lib/systemd/coredump:	default (значение по умолчанию)	
Минимальный объем свободного дискового пространства в байтах.	30G (значение по умолчанию)	
Сохраните в безопасное место и удалите файл /etc/ecss/ssl/ecss10root.key!	Ok (значение по умолчанию)	

Проверить статус сервисов следующей командой:

на обоих хостах

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
```

В случае статуса "active" продолжаем дальше, в случае статуса "inactive" выполнить команду "sudo systemctl restart ecss-<имя пакета>"

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
active
active
active
active
active
```

Установка лицензии

Система ECSS-10 использует ELM-лицензирование – сервис распределения лицензий ECSS-экосистемы, состоит из **ecss-license-agent** и **ecss-license-master** .

- **ECSS License Agent** - Адаптер для работы с EcssLicenseMaster (LM). Получает список хостов EcssLicenseMaster, реализует логику подключения и переподключения к LM.
  - Отвечает за получение, валидацию и дальнейшую отправку лицензий клиенту.
  - Отправляет статусы подключения и полученные лицензии клиенту.
  - Предоставляет API для отправки HTTP запросов и WS событий на LM.
- **Ecss License Master** - Сервис оркестрации параметров лицензии для ECSS-узлов.
  - Отвечает за:
    - Получение лицензии для конкретного ECSS-узла (SSW / AuP) либо устройства (SMG)
    - Распределение параметров лицензии между ECSS-узлами и ECSS-сервисами
    - Контроль активности ECSS-узлов и их валидация во избежание тиражирования лицензии сверх указанных лимитов
    - Контроль изменения указанных лимитов на ECSS-узлах, и их динамическое распределение в соответствии с текущей нагрузкой.

ecss-license-agent является составной частью SSW, входит в пакет ecss-node. Отдельной конфигурации не требует.

В свою очередь **Ecss License Master** может быть установлен как на SSW так и на отдельно выделенный сервер. В случае если Ecss License Master уже установлен на вашем проекте, то потребуется только дополнить запись о новом компоненте в конфигурационном файле LM, и следующий пункт – установка "Установка LM" не потребуется.

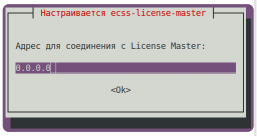
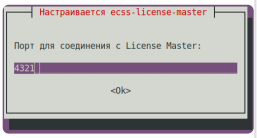
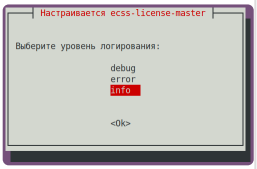
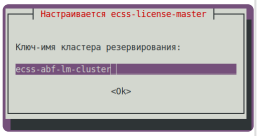
Установка LM на SSW host

Установить пакет LM (ecss-license-master)

если кластер, то на обоих хостах

```
sudo apt install ecss-license-master
```

При установке пакета ecss-license-master будут заданы вопросы:

Вопросы	Ответы	Примеры
Адрес для соединения с License Master: License Master listen ip:	0.0.0.0 (значение по умолчанию)	
Порт для соединения с License Master: License Master listen port:	4321 (значение по умолчанию)	
Выберите уровень логирования: Log level:	info (значение по умолчанию), возможные варианты: debug, error, info	
Ключ-имя кластера резервирования: License Master cluster key:	указать уникальное имя ecss-lm-cluster (значение по умолчанию) Менеджеры с одинаковыми именами попытаются собраться в кластер и резервировать друг друга.	

Данные сохраняются в файле - **/etc/ecss/ecss-license-master/config.env** :

```
ECSS_LM_HOST=0.0.0.0
ECSS_LM_PORT=4321
ECSS_LM_LOG_LEVEL=info
RELEASE_COOKIE=ecss-abf-lm-cluster
```

Конфигурация LM

Сервис ecss-license-master использует два конфигурационных файла: **/etc/ecss/ecss-license-master/config.env** и **/etc/ecss/ecss-license-master/config.yaml**.
В файл **config.env** информация сохраняется в момент установки пакета **ecss-license-master**.
Config.yaml файл содержит основные настройки для подключения к ELM серверу, откуда брать лицензии, какие лицензии, и куда их потом передавать, **все это нужно сконфигурировать вручную**.

При установке LM на SSW

Выполнить команду:

если кластер, то на обоих хостах

```
sudo nano /etc/ecss/ecss-license-master/config.yaml
```

информация в файле после установки

```
elm_addresses: []
licenses: []
ecss_nodes: []
```

В поле **elm_adresses** нужно прописать список серверов ELM к которым мы хотим подключаться для получения лицензий. Первый адрес в списке используется как основной. Дополнительные строки с адресами работают как резерв.

Данные ELM сервера(ов) получить у менеджера проекта.

пример:
- "elm-1.eltex.loc:8099"
- "192.168.111.22:8099"
- "elm-3.eltex.loc:8099"

Дальше задаются сами лицензии в поле **licenses**.

- **id** – может быть любой, главное чтоб был уникальный у каждой лицензии. Влияет только на адрес лицензии на самом LM (например 0,1,2,3).
- **kind** – для SSW поддерживается только ***ssw.***
- **type** – выбирается как ***elm*** если запрашивать данную лицензию нужно с ELM сервера. Если нужно использовать файловую лицензию SSW, то тип выбирается как ***ecss_license***.
  - при ***elm*** типе, необходимо задать поля **license_key** и **product_id** для идентификации лицензии.
  - при ***ecss_license*** типе, задается поле **license** содержащее ключ статичной лицензии

Параметры **license_key** и **product_id** должны быть согласованы с данными загруженными на ELM сервере .

Значение параметров **license_key**, **product_id** или **license**, **passport** нужно получить у менеджера проекта.

Блок **ecss_nodes**.
В нем нужно задать данные сервисов которые будут получать лицензию. В нашем случае SSW.
Сервис задается тремя параметрами:

- **id –** идентификатор сервиса, должен совпадать с ECSS_ID лицензируемого SSW.
- **kind –** тип лицензируемого сервиса. по аналогии с блоком лицензий, для SSW поддерживается только ***ssw.***
- **license_id –** идентификатор лицензии которая будет отправляться на данный сервис. Соответствует идентификаторам лицензии из блока **licenses**

⚠ Два SSW с **разными ECSS_ID** не могут получить одну и ту же лицензию, конфигурация с таким распределением будет помечена некорректной.
Кроме того, два SSW с **одинаковыми ECSS_ID** не смогут одновременно получать одну и ту же лицензию с одного LM,
поэтому крайне важно корректно устанавливать свой ECSS_ID и делать его уникальным как минимум в зоне видимости используемого LM.

✓ Формат **ECSS_ID**
Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире.

```
[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]
```

⚠ Для SSW работающего в кластере указываем только одну запись в блок **ecss_nodes** на кластер, индивидуально для каждого хоста указывать нельзя. Иначе будут генерироваться два запроса с одинаковым ECSS_ID что недопустимо.

Пример: файла config.yaml

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
   kind: SSW
   type: elm
   license_key: "ssw1test"
   product_id: "ECSS1000001"
 - id: 1
   kind: ssw
   type: ecss_license
   license: "af615ebb92d381125ff"
ecss_nodes:
 - id: abf.test
   kind: ssw
   license_id: 0
```

Выполнить перезагрузку сервиса **ecss-license-master.service** для обновления конфигурации из файла **/etc/ecss/ecss-license-master/config.yaml** следующей командой:

если кластер, то на обоих хостах

```
sudo systemctl restart ecss-license-master.service
```

При подключении SSW к ранее установленному LM

В случае если LM уже установлен и используется в локальной сети требуется добавить параметры нового SSW в конфигурационный файл **/etc/ecss/ecss-license-master/config.yaml** в блоках licenses и ecss_nodes.

Выполнить команду:

если кластер, то на обоих хостах

```
sudo nano /etc/ecss/ecss-license-master/config.yaml
```

для примера:

Было:

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
   kind: SSW
   type: elm
   license_key: "ssw1test"
   product_id: "ECSS1000001"
 - id: 1
   kind: ssw
   type: ecss_license
   license: "af615ebb92d381125ff"
 - id: 2
   kind: SSW
   type: elm
   license_key: "ssw2test"
   product_id: "ECSS1000002"
ecss_nodes:
 - id: 1.test
   kind: ssw
   license_id: 0
 - id: 2.test
   kind: ssw
   license_id: 1
 - id: 3.test
   kind: ssw
   license_id: 2
```

Стало:

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
```

```
    kind: SSW
    type: elm
    license_key: "ssw1test"
    product_id: "ECSS1000001"
  - id: 1
    kind: ssw
    type: ecss_license
    license: "af615ebb92d381125ff"
  - id: 2
    kind: SSW
    type: elm
    license_key: "ssw2test"
    product_id: "ECSS1000002"
  - id: 3
    kind: SSW
    type: elm
    license_key: "ssw3test"
    product_id: "ECSS1000003"
ecss_nodes:
  - id: 1.test
    kind: ssw
    license_id: 0
  - id: 2.test
    kind: ssw
    license_id: 1
  - id: 3.test
    kind: ssw
    license_id: 2
  - id: abf.test
    kind: ssw
    license_id: 3
```

Выполнить перезагрузку сервиса **ecss-license-master.service** для обновления конфигурации из файла **/etc/ecss/ecss-license-master/config.yaml** следующей командой:

если кластер, то на обоих хостах

```
sudo systemctl restart ecss-license-master.service
```

Подключение SSW к Менеджера Лицензий

Выполнить конфигурацию подключения SSW к LM следующей командой в CoCon, (в примере LM установлен в кластере, IPadd хостов 10.0.10.41 и 10.0.10.42. значение порта из файла /etc/ecss/ecss-license-master/config.env по умолчанию 4321):

выполняется в CoCon

```
/system/licence/manager/set --hosts [https://10.0.10.41:4321, https://10.0.10.42:4321]
```

В случае установки LM на хостах SSW указывать IP адреса OAM

После выполнения проверить статус подключения командой:

выполняется в CoCon

```
/system/licence/manager/show-status
┌──────────────────────────┬───────┬─────┐
│          Host            │Current│Alive│
├──────────────────────────┼───────┼─────┤
│https://10.0.10.41:4321│*      │true │
│https://10.0.10.42:4321│       │true │
└──────────────────────────┴───────┴─────┘
```

Все хосты должны иметь статус alive=true. Один должен быть "current".

- Current - отображает к какому хосту подключён SSW. Если есть подключение, указывается *, если нет - ничего не указано.
- Alive - отображает статус доступности хоста (healthcheck).

Подключение к License Manager

Отсутствие подключения к License Manager \ ELM будет равноценно отсутствию доступа к токену, что приведёт к критическим авариям, переходу в аварийный режим по истечению которого, если связь не будет восстановлена, система перейдёт на дефолтную лицензию.

данное поведение применимо только при работе в **elm** режиме

Если статус к одному из хостов LM показан current, можно посылать запрос на загрузку лицензии на SSW командой:

выполняется в CoCon

```
/cluster/storage/ds1/licence/request
```

пример выполнения команды /cluster/storage/ds1/licence/request

type=elm

```
/cluster/storage/ds1/licence/request
Licence received
[*******                                                    ] 6s 2ms
Success: Licence parameters applied
```

В режиме "type=ecss_license" команда **/cluster/storage/ds1/licence/request** выполнится только в случае если был предварительно загружен паспорт.

type=ecss_license

```
/cluster/storage/ds1/licence/request
Waiting for licence...
[*******                                                    ] 6s
```

A	Description	Old Value	New Value
^	Support for reservation of call-processes	false	true
^	Maximum call duration (in seconds)	60	2678400
^	Maximum number of simultaneous calls	5	10000
^	Elph		
^	total count of members of the one Elph group	10	100
^	Subscribers limit	10	500000
^	Virtual subscribers limit	infinity	1000
^	Add-on conferences		
^	total count	30	1000
^	Add-on conferences		
^	members of the one add-on conference	16	300
^	Chat rooms		
^	total count	30	100
^	Chat rooms		
^	members of the one chatroom	16	300
^	Meet Me		
^	total count	0	100
^	Meet Me		
^	members of the one Meet Me conference	0	300
^	SORM		
^	enabled	false	true
+	channels on SORM mediator		64
^	Sorm extractor	none	mfi
^	Support antifraud system	none	custom
+	Call center		
+	operator's capability for look at calls in queue		true

```
|+| operator's capability for use Intervension SS       |         |true
              |
|^|Call center                                          |         |
              |
|^| active agents                                       |0        |1000
              |
|^|Call center                                          |         |
              |
|^| active supervisors                                  |0        |100
              |
|^|Teleconference                                       |         |
              |
|^| channels                                            |3        |2000
              |
|^|Teleconference                                       |         |
              |
|^| max members of one teleconference                   |0        |200
              |
|^|Teleconference                                       |         |
              |
|^| active count                                        |0        |32
              |
|^|TSMN system                                          |         |
              |
|^| concurrent calls (active)                           |0        |50
              |
|^|TSMN system                                          |         |
              |
|^| concurrent calls (backup)                           |0        |50
              |
|^|TSMN system                                          |         |
              |
|^| backup mode                                         |none     |"backup"
              |
|^|Total count of simultaneous records voice calls      |0        |200
            |
|+|IVR                                                  |         |
              |
|+| customization enabled                               |         |true
              |
|+| Automatic Speech Recognition (ASR) subsystem enabled|         |true
              |
|^|Channels on dialer outgoing calls                    |2        |5
            |
|+|Supplementary Services licence package(s)            |         |
              |
|+| name                                                |         |'ECSS-ADV'
              |
|+| limit                                               |         |10000
            |
|+| description                                         |         |"Additional services of 4
and 5 levels" |
|+| SS list                                             |         |[1,2,3,4,5]
              |
|+| name                                                |         |'ECSS-BAS'
              |
|+| limit                                               |         |10000
            |
```

```
|+| description                                          |         |"Basic services of 1 and
2 levels"      |
|+| SS list                                              |         |[1,2]
               |
|+| name                                                 |         |'ECSS-BAS+'
               |
|+| limit                                                |         |10000
               |
|+| description                                          |         |"Expansion of basic
services of 3 level"|
|+| SS list                                              |         |[1,2,3]
               |
|^|GEO backup                                           |         |
               |
|^| enabled                                            |false    |true
               |
|^|SIGTRAN                                             |         |
               |
|^| SEP                                                |false    |true
               |
|^| STP                                                |false    |true
               |
└─┴──────────────────────────────────────────────────────┴─────────┴──────────────────────
──────────────┘
Legend:
    '^' - Changed;
    '-' - Deleted;
    '+' - New.


[request] You are trying to add a license that is different from the current one by the pa
          rameters in the table (other values will remain unchanged).
Licence received      o ?> yes
[*******                                                          ] 1m 15s
Success: Licence parameters applied
```

В зависимости от выбранного типа лицензии для SSW : "type=**elm**" или "type=**ecss_license**" на команду просмотра лицензий получим разный вывод:

type=ecss_license

```
/cluster/storage/ds1/licence/list-licence
┌──┬──────────────────────┬──────────────┬──────┬──────────────────┬──────────────────────┬──
──────────────────┬──────────┐
|Id|Creation date(UTC) |   SSW ID   |Active|  Description   |Comm. Exp.
Date(UTC)|Expiration date(UTC)|Time left|
├──┼──────────────────────┼──────────────┼──────┼──────────────────┼──────────────────────┼──
──────────────────┼──────────┤
|1 |12.03.2026 18:40:00|ECSS 010070 |*     |ECSS TPM License|13.03.2027 06:59:59
|14.03.2026 10:44:05 |1d 6h 59m|
|0 |01.01.1990 00:00:00|ECSS DEFAULT|      |Default licence |                      |
|          |
```

```
└──┴──────────────────────┴──────────────┴──────┴──────────────────┴──────────────────────────┴──────
──────────────────┴────────────┘
```

TPM - является полным аналогом лицензирования которое использовалось раньше, с токеном и паспортом, без подключения к LM.

или

type=elm

```
/cluster/storage/ds1/licence/list-licence
┌──┬──────────────────────┬──────────────┬──────┬──────────────────┬──────────────────────────┬──────
──────────────────┴────────────┐
|Id|Creation date(UTC) |   SSW ID   |Active|  Description   |Comm. Exp.
Date(UTC)|Expiration date(UTC)|Time left|
├──┼──────────────────────┼──────────────┼──────┼──────────────────┼──────────────────────────┼──────
──────────────────┴────────────┤
|1 |03.02.2026 16:26:34|ECSS2000009 |*     |ECSS ELM License|03.02.2027 16:26:34
|14.03.2026 10:49:12 |1d 6h 59m|
|0 |01.01.1990 00:00:00|ECSS DEFAULT|      |Default licence |                          |
|          |
└──┴──────────────────────┴──────────────┴──────┴──────────────────┴──────────────────────────┴──────
──────────────────┴────────────┘
```

Так же требуется выполнить конфигурацию для Sip адаптера , выполнив следующую команду:

на обоих хостах

```
sudo systemctl edit ecss-pa-sip.service
```

Добавить в конфигурационный файл следующие данные:

```
[Service]
LimitNOFILE=65536
```

Перезагрузите конфигурацию

на обоих хостах

```
sudo systemctl daemon-reload
```

Актуализировать конфигурационный файл, выполнив перезагрузку сервера:

```
sudo reboot
```

Установка остальных пакетов ecss и их настройка

Для корректной работы кластерной конфигурации необходимо настроить работу **RestFS** на базе **GlusterFS-сервера.**

Следует проверить в файле /etc/dnsmasq.d/ecss-broker, что адреса **primary и secondary broker** соответствуют указанным при инсталляции пакета **ecss-node**.

на обоих хостах

```
cat /etc/dnsmasq.d/ecss-broker
```

Пример содержания файла на **ecss1** и **ecss2** (содержимое файлов должно быть **одинаковым** на обоих серверах):

на обоих хостах

```
address=/primary.broker.ecss/10.0.10.41
address=/secondary.broker.ecss/10.0.10.42
```

glusterfs

Выполните настройку glusterfs для ecss-restfs на **первом хосте (ecss1)**, для этого **установите** пакеты glusterfs-server и attr **на обоих** хостах:

на обоих хостах

```
sudo apt install glusterfs-server attr
```

Запустите glasterd на **обоих** хостах кластера:

на обоих хостах

```
sudo systemctl start glusterd
```

После запуска выполните создание связи с удаленным виртуальным хостом, выполните команду на **ecss1**:

на ecss1

```
sudo gluster peer probe 10.0.10.42
```

Проверьте наличие созданной связи, выполните команду на **ecss2**:

на ecss2

```
sudo gluster peer status
```

Должна появиться информация о первом хосте — ecss1:

на ecss2

```
Number of Peers: 1

Hostname: ecss1
Uuid: d7319720-ea3a-43a4-a7d5-e97bf9d205b9
State: Peer in Cluster (Connected)
```

Выполните аналогичную проверку на другом хосте — **ecss1**. Вы также должны увидеть информацию о партнере **ecss2** (или его IP-адрес).

на ecss1

```
sudo gluster peer status
Number of Peers: 1

Hostname: 10.0.10.42
Uuid: f405dbb1-4f6d-4f2f-a116-f56197d18788
State: Peer in Cluster (Connected)
```

Создайте кластер для репликации, запустите репликацию и посмотрите ее статус, выполните следующие команды на **ecss1**:

на ecss1

```
sudo gluster volume create ecss_volume replica 2 transport tcp 10.0.10.41:/var/lib/ecss/
glusterfs 10.0.10.42:/var/lib/ecss/glusterfs force
```

на ecss1

```
sudo gluster volume start ecss_volume
```

на обоих хостах

```
sudo gluster volume info
```

Таким образом, статус репликации будет выглядеть следующим образом, необходимо обратить внимание на поля "**Status**" и "**Bricks**" — они должны иметь следующий вид:

```
Volume Name: ecss_volume
Type: Replicate
Volume ID: 3bfc7587-0f85-48ed-9612-21f0d79c6e52
Status: Started
Snapshot Count: 0
Number of Bricks: 1 x 2 = 2
Transport-type: tcp
Bricks:
Brick1: 10.0.10.41:/var/lib/ecss/glusterfs
Brick2: 10.0.10.42:/var/lib/ecss/glusterfs
Options Reconfigured:
transport.address-family: inet
nfs.disable: on
performance.client-io-threads: off
```

Чтобы смонтировать glusterfs раздел, выполните на **обоих** хостах **ecss1** и **ecss2** следующие действия:

- Создайте новый systemd unit

на обоих хостах

```
sudo nano /etc/systemd/system/ecss-glusterfs-mount.service
```

- Добавьте в него следующую конфигурацию:

```
[Unit]
Description=mount glusterfs
After=network.target
Requires=network.target

[Service]
RemainAfterExit=no
Type=forking
RestartSec=10s
Restart=always
ExecStart=/sbin/mount.glusterfs localhost:/ecss_volume /var/lib/ecss/restfs -o fetch-
attempts=10
ExecStop=/bin/umount /var/lib/ecss/restfs

[Install]
WantedBy=multi-user.target
```

- Добавить unit в автозагрузку на **обоих** хостах.

на обоих хостах

```
sudo systemctl enable  glusterd.service
```

на обоих хостах

```
sudo systemctl enable  ecss-glusterfs-mount.service
```

- Создайте директорию для корректной установки RestFS на **обоих** хостах:

на обоих хостах

```
sudo mkdir /var/lib/ecss/restfs
```

- Если хосты не могут быть перезагружены, то можно выполнить следующие команды на **обоих** хостах:

на обоих хостах

```
sudo systemctl daemon-reload
```

на обоих хостах

```
sudo systemctl restart ecss-glusterfs-mount.service
```

Проверьте, что раздел под монтировался, для этого на обоих хостах выполните команду - **df -h**:

на обоих хостах

```
df -h
```

на обоих хостах

```
df -h

/dev/sda8                     4,5G  213M  4,1G   5% /var/log
/dev/sda5                      37G   48M   35G   1% /var/lib/ecss/ecss-media-server/
records
/dev/sda6                      19G   44M   18G   1% /var/lib/ecss/cdr
/dev/sda7                      19G   44M   18G   1% /var/lib/ecss/statistics
```

```
/dev/sda9                       19G  7,6G  9,7G  44% /var/log/ecss
localhost:/ecss_volume          46G   59M   44G   1% /var/lib/ecss/restfs
```

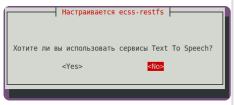
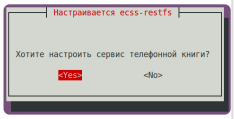
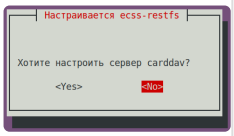
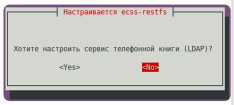
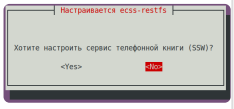
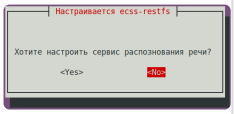
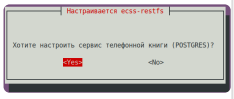
При просмотре информации должен появиться подмонтированный раздел — **localhost:/ecss_volume** 46G 59M 44G 1% **/var/lib/ecss/restfs**

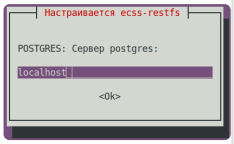
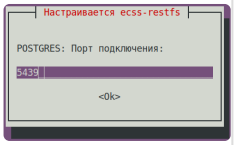
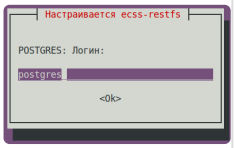
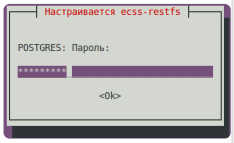
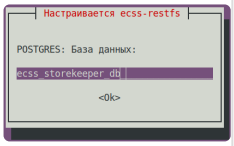
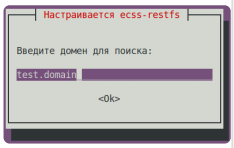
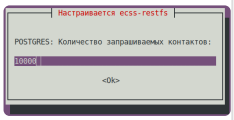
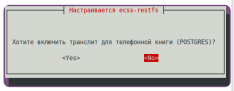
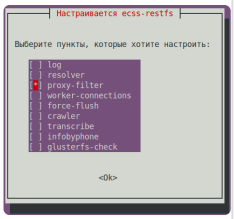
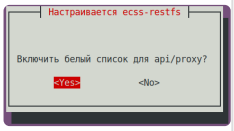
Установка пакета ecss-restfs и его настройка

установить на ecss1, затем на ecss2

```
sudo apt install -y ecss-restfs
```

Установка **ecss-restfs.** В ходе установки нужно будет ответить на ряд вопросов для создания необходимых конфигурационных файлов. Также инсталлятор предложит установить и настроить пакет Text2speech от Yandex.

Вопросы ecss-restfs	Ответы	Пример
Хотите ли вы использование функции Text To Speeh (Use TTS service)	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (Configure phone book)	**Yes** (нужно указать)	
Хотите настроить сервер carddav?	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (LDAP)?	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (SSW)?	No (значение по умолчанию)	
Хотите настроить сервис распознавания речи (Configure speech recognition service)	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (POSTGRES)?	Yes (значение по умолчанию)	

Вопросы ecss-restfs	Ответы	Пример
POSTGRES: Сервер postgres:	localhost (значение по умолчанию)	
POSTGRES: Порт подключения:	5439 (значение по умолчанию)	
POSTGRES: Логин:	postgres (значение по умолчанию)	
POSTGRES: Пароль:	postgres1 (значение по умолчанию)	
POSTGRES: База данных:	ecss_storekeeper_db (значение по умолчанию)	
Введите домен для поиска:	test.domain (значение по умолчанию)	
POSTGRES: Количество запрашиваемых контактов:	10000 (значение по умолчанию)	
Хотите включить транслит для телефонной книги (LDAP)?	No (значение по умолчанию)	
Выберите пункты, которые хотите настроить:	**proxy-filter** (необходимо выбрать)	
Включить белый список для api/proxy?		

белый список адресов прописать в файл /etc/ecss/ecss-restfs/api-proxy.conf

```
# ------------------------------------------------------------
# Формат файла фильтра API (api/proxy):
#   - IP: 192.168.0.1
#   - Домен: example.com
#   - Wildcard: *.wildcard.com
#   - Комментарии: строки, начинающиеся с #
# Пример:
#   192.168.0.1
#   example.com
#   # комментарий
#   *.wildcard.com
# ------------------------------------------------------------
```

ВАЖНО!

В случае успешного проксирование (200) кеш хранится 28 дней, поэтому для применения конфигурации фильтра, в случае изменения, - кеш (/var/cache/ecss/restfs-api) стоит удалить и перезапустить сервис ecss-restfs.

после установки пакета ecss-restfs проверьте наличие wav файлов авто-информатора в директории /var/lib/ecss/restfs/system/sounds/ командой:

на обоих хостах

```
ls -la /var/lib/ecss/restfs/system/sounds/
```

наличие wav файлов в указной выше директории говорит о корректности установки пакета ecss-restfs.

Проверить доступность этих файлов извне, для этой проверки выполним поочередно команду на **ecss1** и **ecss2**:

```
wget http://ecss1:9990/system/sounds/ai_you.wav
```

```
wget http://ecss2:9990/system/sounds/ai_you.wav
```

пример

```
wget http://ecss1:9990/system/sounds/ai_you.wav
--2025-09-16 08:27:36--  http://ecss1:9990/system/sounds/ai_you.wav
Распознаётся ecss1 (ecss1)… 10.0.10.41
Подключение к ecss1 (ecss1)|10.0.10.41|:9990... соединение установлено.
HTTP-запрос отправлен. Ожидание ответа… 200 OK
Длина: 11670 (11K) [audio/x-wav]
Сохранение в: «ai_you.wav»
```

```
ai_you.wav.1                                        100%
[=============================================================================
===>]  11,40K  --.-KB/s    за 0s

2025-09-16 08:27:36 (81,5 MB/s) - «ai_you.wav» сохранён [11670/11670]

wget http://ecss2:9990/system/sounds/ai_you.wav
--2025-09-16 08:27:44--  http://ecss2:9990/system/sounds/ai_you.wav
Распознаётся ecss2 (ecss2)… 10.0.10.42
Подключение к ecss2 (ecss2)|10.0.10.42|:9990... соединение установлено.
HTTP-запрос отправлен. Ожидание ответа… 200 OK
Длина: 11670 (11K) [audio/x-wav]
Сохранение в: «ai_you.wav.1»

ai_you.wav.2                                        100%
[=============================================================================
===>]  11,40K  --.-KB/s    за 0s

2025-09-16 08:27:44 (272 MB/s) - «ai_you.wav.1» сохранён [11670/11670]
```

проверочные файлы можно удалить

```
rm ai_you.wav
rm ai_you.wav.1
```

если SSW работает в кластере

В случае кластерной конфигурации, для предотвращения одновременной обработки/конвертации/удаления одного файла (.pcm/.wav/.mp3) на разных серверах ecss1/ecss2 → сервис **ecss-restfs-crawler** должен быть запущен **только на одном из серверов** кластера. Для этого на сервере ecss2 выполнить следующие команды:

на ecss2

```
sudo systemctl disable ecss-restfs-crawler.timer
```

на ecss2

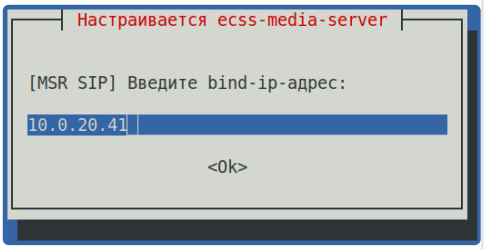
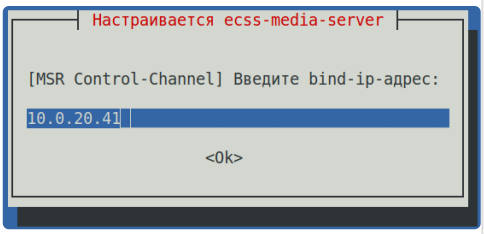
```
sudo systemctl stop ecss-restfs-crawler.timer
```

Установка media и web пакетов ecss и их настройка

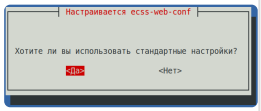
на обоих хостах

```
sudo apt install -y ecss-media-server ecss-web-conf
```

Для медиасервера (*ecss-media-server/MSR*) возможно начальное конфигурирование с записью параметров в файл конфигурации,
для этого нужно провести конфигурацию **transport bind-addr,mcc bind-addres**:

Вопросы ecss-media-server	Ответы для ecss1	Ответы для ecss2	Пример
[MSR SIP] Введите bind-ip адрес (Enter)	**10.0.20.41** (необходимо ввести)	**10.0.20.42** (необходимо ввести)	
[MSR Control-Channel] Введите bind-ip-адрес	**10.0.20.41** (необходимо ввести)	**10.0.20.42** (необходимо ввести)	

Web-конфигуратор позволяет сделать управление системой более наглядным и комфортным. Установка web-конфигуратора не является обязательной, но рекомендуется.
Также при установке пакета **ecss-web-conf** автоматически устанавливается пакет **ecss-subsriber-portal-ui**. Приложение "Портал абонента" системы ECSS-10 позволяет абонентам системы самостоятельно управлять услугами, просматривать информацию по совершенным вызовам, активным конференциям, а также настраивать собственные IVR-скрипты для входящих вызовов. Описание работы веб-конфигуратора приведено в разделе "Портал абонента".

Вопросы ecss-web-conf	Ответы для ecss1/ecss2	Пример
Хотите ли вы использовать стандартные настройки?	Yes (значение по умолчанию)	

После формирования конфигураций по умолчанию переходим в директорию, в которой находятся файлы конфигурации и производим их проверку:

```
cd /etc/ecss/ecss-media-server/
```

```
cat /etc/ecss/ecss-media-server/config.xml
```

Конфигурация для msr: файл - config.xml, в директории conf.d файл default.xml.
По своей сути default.xml — это дополнение config.xml, которое определяет секцию аккаунтов. Это сделано для того, чтобы после обновлений пакета данная конфигурация оставалась неизменной. Вид config.xml представлен здесь: файл конфигурации.

```
<?xml version="1.0" encoding="utf-8"?>
<config date="02:30:33 07.12.2023">
```

```
  <general log-level="3" log-rotate="yes" max-calls="8192" max-in-group="512" load-sensor="medi
a" load-delta="10" calls-delta="100" spool-dir-size="100M" log-name="msr.log" log-path="/var/
log/ecss/media-server" use-srtp="disabled" enable-ice-transport="no" ice-update="no"
 aggressive-ice="yes" stun-server="" suspicious-mode="no"/>
  <transport bind-addr="10.0.20.41" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca" user="webrtc@live.com" password="muazkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60" rtcp-timeout="0" rtp-
timeout="350" udp-src-check="no" cn-multiplier="3" port-start="12000" port-range="2048" tias-
in-sdp="no" thread-cnt="2" vid-enc-threads="2" vid-dec-threads="2" video-conf-layout="evenly"
 keyframe-interval="1000" vid-decode-delay="100" silence-threshold="-30" dtmf-flash-disable="no
" video-dscp="0" other-dscp="0" dummy-video-src="/usr/share/ecss-media-server/video/
dummy_video.yuv" video-enc-width="640" video-enc-height="360" finalsilence="1000" rtcp-stat-
dump="yes" silent-codec-switch="yes"/>
  <codec pcma="1" pcmu="2" ilbc="0" gsm="0" g722="3" g726="0" g729="0" speex="0" l16="0" g7221="
0" opus="0" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
  <accounts>
    <!-- <dynamic msr_name="msr.name"
            realm="sip:127.0.0.1:5000"
            dtmf_mode="rfc+inband+info"
            auth_name="user"
            auth_password="password" /> -->
  </accounts>
  <pbyte>
    <mcc bind-addr="10.0.20.41" port="5700"/>
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

По умолчанию после инсталляции ECSS-10 **активизирует только** кодеки — **pcma, pcmu, g722,** h264, h263-1998, t38. Остальные имеют статус =0 = выкл. **Если требуется** активировать остальные кодеки, в редакторе nano измените приоритет с 0 на цифру приоритета (уровень приоритета не должен повторятся для сервиса audio/video/fax)
sudo nano /etc/ecss/ecss-media-server/config.xml

чтобы активизировать **все** кодеки
строчку -
<codec pcma="1" pcmu="2" ilbc="**0**" gsm="**0**" g722="3" g726="**0**" g729="**0**" speex="**0**" l16="**0**" g7221="**0**" opus="**0**" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
привести к виду -
<codec pcma="1" pcmu="2" ilbc="**6**" gsm="**4**" g722="3" g726="**11**" g729="**5**" speex="**7**" l16="**8**" g7221="**9**" opus="**10**" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>

Недопустимо указывать у разных кодеков одинаковый приоритет, для примера g722="5" g729="5". В этом случае media-server не сможет работать корректно.
За исключением "0" - выключено.

Рассмотрим секцию аккаунтов (файл default.xml):

После инсталляции пакетов в фале будет указан ip-add 127.0.0.1 , и только для одного хоста.

```
cat /etc/ecss/ecss-media-server/conf.d/default.xml
```

```
<?xml version="1.0"?>
<config>
      <accounts>
            <dynamic msr_name="msr.ecss1" realm="sip:127.0.0.1:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>
      </accounts>
</config>
```

Требуется указать в конфигурации для обоих хостов, реальные IP-add.

Необходимо привести содержимое файла к виду указанному снизу (в примере ecss1 имеет IP-add 10.0.20.41/ ecss2 - 10.0.20.42):

на обоих хостах

```
sudo nano /etc/ecss/ecss-media-server/conf.d/default.xml
```

Настройка msr для ecss1 (/etc/ecss/ecss-media-server/conf.d/default.xml)

```
<?xml version="1.0"?>
<config>
  <accounts>
    <dynamic msr_name="msr.ecss1" realm="sip:10.0.20.41:5000" dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password">
      <via dynamic-ifaces="no">
        <iface name="net.20" ip="10.0.20.41"/>
      </via>
    </dynamic>
    <dynamic msr_name="msr.ecss1" realm="sip:10.0.20.42:5000" dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password">
      <via dynamic-ifaces="no">
        <iface name="net.20" ip="10.0.20.42"/>
      </via>
    </dynamic>
  </accounts>
</config>
```

Настройка msr для ecss2 (/etc/ecss/ecss-media-server/conf.d/default.xml)

```
<?xml version="1.0"?>
<config>
  <accounts>
    <dynamic msr_name="msr.ecss2" realm="sip:10.0.20.42:5000" dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password">
      <via dynamic-ifaces="no">
        <iface name="net.20" ip="10.0.20.42"/>
      </via>
    </dynamic>
    <dynamic msr_name="msr.ecss2" realm="sip:10.0.20.41:5000" dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password">
      <via dynamic-ifaces="no">
        <iface name="net.20" ip="10.0.20.41"/>
      </via>
    </dynamic>
  </accounts>
</config>
```

Указываем текущие настройки, согласно которым происходит регистрация msr на core.

Основными параметрами здесь являются: msr_name и realm:

- msr_name — параметр, определяющий название msr. (рекомендуется задавать msr. и к какому хосту он принадлежит, например msr.ecss1);
- realm — определяет адрес для регистрации на ядре. Точка входа по умолчанию — порт 5000.

После изменения конфигурации файлов **/etc/ecss/ecss-media-server/config.xml** и **/etc/ecss/ecss-media-server/conf.d/default.xml** необходимо выполнить перезапуск сервиса **ecss-media-server**, чтобы изменения вступили в силу, следующей командой:

на обоих хостах

```
sudo systemctl restart ecss-media-server
```

Настройка VRRP для SIP-адаптера

Для Astra Linux

Если ECSS-10 установлен на Astra Linux, то необходимо изменить скрипт проверки доступности контрольного SIP-порта **/usr/bin/ecss_pa_sip_port:**

на обоих хостах

```
#!/bin/bash
netstat -plane | grep $1
#nc -z 127.0.0.1 $1
```

Выполните настройку VRRP для SIP-адаптера, для этого на **обоих** хостах создайте файлы следующего содержания:

Глобального конфигурационного файла **keepalived.conf**. На обоих хостах содержимое файла **одинаковое**:

на обоих хостах

```
sudo nano /etc/keepalived/keepalived.conf
```

Скопируйте и вставьте следующий текст в файл /etc/keepalived/keepalived.conf

на обоих хостах

```
global_defs {
    vrrp_version 3          # версия протокола VRRP (2 или 3)
    script_user nobody      # ограниченный в правах системный пользователь, от которого будут
запускаться скрипты проверки доступности
    enable_script_security  # не запускать скрипты от root, если часть пути к ним доступна на
запись для обычных пользователей
}
```

```
include /etc/keepalived/sip.conf
```

Выполните настройку VRRP для SIP-адаптера, для этого на **обоих** хостах создайте файлы следующего содержания:

на обоих хостах

```
sudo nano /etc/keepalived/sip.conf
```

etc/keepalived/sip.conf для ecss1

на ecss1

```
vrrp_script check_sip {
    script "/usr/bin/ecss_pa_sip_port 65535"
    interval 2
    timeout 2
}

# Конфигурация адреса для первого
виртуального адреса SIP-адаптера
vrrp_instance SIP1 {
    state MASTER                    # Исходное
состояние при старте
    interface net.20                # Имя
сетевого интерфейса, на котором будет
работать протокол VRRP
    virtual_router_id 31            #
Уникальный идентификатор роутера (0..255)
    priority 100                    #
Приоритет (0..255) чем выше - тем больше
    advert_int 1                    # Интервал
рассылки уведомлений (с)
    preempt_delay 60                # Интервал
ожидания мастера при старте демона (с) при
исходном состоянии BACKUP

    unicast_src_ip 10.0.20.41      #
Собственный реальный IP-адрес
    unicast_peer {
        10.0.20.42                  # Реальный
 IP-адрес соседа
    }

    virtual_ipaddress {
        # Виртуальный IP-адрес и маска
        # dev - сетевой интерфейс, на
котором будет поднят виртуальный адрес
        # label - метка виртуального
интерфейса (для удобства идентификации)
        10.0.20.31/24 dev net.20 label
net.20:SIP1
    }

    track_script {
        check_sip
    }
}

# Конфигурация адреса для второго
виртуального адреса SIP-адаптера

vrrp_instance SIP2 {
```

etc/keepalived/sip.conf для ecss2

на ecss2

```
vrrp_script check_sip {
    script "/usr/bin/ecss_pa_sip_port 65535"
    interval 2
    timeout 2
}

# Конфигурация адреса для первого
виртуального адреса SIP-адаптера
vrrp_instance SIP1 {
    state BACKUP                    # Исходное
состояние при старте
    interface net.20                # Имя
сетевого интерфейса, на котором будет
работать протокол VRRP
    virtual_router_id 31            #
Уникальный идентификатор роутера (0..255)
    priority 50                     #
Приоритет (0..255) чем выше - тем больше
    advert_int 1                    # Интервал
рассылки уведомлений (с)
    preempt_delay 60                # Интервал
ожидания мастера при старте демона (с) при
исходном состоянии BACKUP

    unicast_src_ip 10.0.20.42      #
Собственный реальный IP-адрес
    unicast_peer {
        10.0.20.41                  # Реальный
 IP-адрес соседа
    }

    virtual_ipaddress {
        # Виртуальный IP-адрес и маска
        # dev - сетевой интерфейс, на
котором будет поднят виртуальный адрес
        # label - метка виртуального
интерфейса (для удобства идентификации)
        10.0.20.31/24 dev net.20 label
net.20:SIP1
    }

}

# Конфигурация адреса для второго
виртуального адреса SIP-адаптера

vrrp_instance SIP2 {
    state MASTER                        #
Исходное состояние при старте
```

etc/keepalived/sip.conf для ecss1

```
    state BACKUP                        #
Исходное состояние при старте
    interface net.20                    # Имя
сетевого интерфейса, на котором будет
работать протокол VRRP
    virtual_router_id 32                #
Уникальный идентификатор роутера (0..255)
    priority 50                         #
Приоритет (0..255) чем выше - тем больше
    advert_int 1                        #
Интервал рассылки уведомлений (с)
    preempt_delay 60                    #
Интервал ожидания мастера при старте демона
 (с) при исходном состоянии BACKUP

    unicast_src_ip  10.0.20.41          #
Собственный реальный IP-адрес
    unicast_peer {
         10.0.20.42                     #
Реальный IP-адрес соседа
    }

    virtual_ipaddress {
        # Виртуальный IP-адрес и маска
        # dev - сетевой интерфейс, на
котором будет поднят виртуальный адрес
        # label - метка виртуального
интерфейса (для удобства идентификации)
        10.0.20.32/24 dev net.20 label
net.20:SIP2
    }

}
```

etc/keepalived/sip.conf для ecss2

```
    interface net.20                    # Имя
сетевого интерфейса, на котором будет
работать протокол VRRP
    virtual_router_id 32                #
Уникальный идентификатор роутера (0..255)
    priority 100                        #
Приоритет (0..255) чем выше - тем больше
    advert_int 1                        #
Интервал рассылки уведомлений (с)
    preempt_delay 60                    #
Интервал ожидания мастера при старте демона
 (с) при исходном состоянии BACKUP

    unicast_src_ip  10.0.20.42          #
Собственный реальный IP-адрес
    unicast_peer {
         10.0.20.41                     #
Реальный IP-адрес соседа
    }

    virtual_ipaddress {
        # Виртуальный IP-адрес и маска
        # dev - сетевой интерфейс, на
котором будет поднят виртуальный адрес
        # label - метка виртуального
интерфейса (для удобства идентификации)
        10.0.20.32/24 dev net.20 label
net.20:SIP2
    }

    track_script {
        check_sip
    }
}
```

В данном случае была добавлена поддержка виртуальных интерфейсов со связью Master - Backup. Для ***ecss1*** основным будет являться ***net.20:SIP1,*** а запасным — ***net.20:SIP2*** соответственно. Важно учесть, что в конфигурации учтено использование адресных переменных. В секции interface требуется указать, на каком интерфейсе будут слушаться VRRP-сообщения, а в секции virtual_ipaddress требуется указать интерфейс, на котором будет восстановлен виртуальный адрес.

Выполните перезапуск keepalived:

на обоих хостах

```
sudo systemctl restart keepalived.service
```

для проверки правильности установки выполните команду ifconfig, в конце выходного сообщения вы должны видеть новый интерфейс - net.20:SIP1 с выбранным IP-адресом

```
ifconfig

. . .
net.20:SIP1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 10.0.20.31 netmask 255.255.255.0 broadcast 0.0.0.0
ether 08:00:27:6d:4f:02 txqueuelen 1000 (Ethernet)
```

Дальнейшая настройка

snpmd

Настройка SNMP описана в разделе "Управление и мониторинг по протоколу SNMP".

Настройка безопасности. SSH

Выполните настройку сервера SSH:

на обоих хостах

```
sudo nano /etc/ssh/sshd_config
```

Укажите в конфигурационном файле порт и адрес, по которому можно обратиться к серверу:

Настройка ssh для ecss1 (/etc/ssh/sshd_config)	Настройка ssh для ecss2 (/etc/ssh/sshd_config)
# This is the sshd server system-wide configuration file. See # sshd_config(5) for more information. # This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin # The strategy used for options in the default sshd_config shipped with # OpenSSH is to specify options with their default value where # possible, but leave them commented. Uncommented options override the # default value. **Port 2000** #AddressFamily any **ListenAddress 10.0.10.11** #ListenAddress :: <...>	# This is the sshd server system-wide configuration file. See # sshd_config(5) for more information. # This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin # The strategy used for options in the default sshd_config shipped with # OpenSSH is to specify options with their default value where # possible, but leave them commented. Uncommented options override the # default value. **Port 2000** #AddressFamily any **ListenAddress 10.0.10.12** #ListenAddress :: <...>

Перезапустите ssh:

на обоих хостах

```
sudo systemctl restart ssh.service
```

Перезагрузка обоих хостов

на обоих хостах

```
sudo reboot
```

Добавление услуг

Установка всех услуг указных в лицензии:

выполняется в CoCon

```
/cluster/storage/ds1/ss/install ds1@ecss1 ss_*
```

Результат:

```
/cluster/storage/ds1/ss/install ds1@ecss1 ss_*
Successfully installed: /var/lib/ecss/ss/ss_dnd.xml
Successfully installed: /var/lib/ecss/ss/ss_clir.xml
...
Successfully installed: /var/lib/ecss/ss/ss_chunt.xml
```

12.6 Первоначальная конфигурация системы с помощью web-конфигуратора

пример первоначальной конфигурации через Web интерефейс

12.6.1 Исходные данные

- Установка системы завершена;
- Система готова к дальнейшей настройке;
- Интерфейсы запущены.

Рекомендуется использовать последние доступные версии браузеров. Рекомендуемые к использованию браузеры: Яндекс, Chrome.

Для начала конфигурации системы необходимо зайти в web-конфигуратор по адресу http://<Ipadd SSW>.

Для определения и регистрации в системе планируются:

- Абоненты с номерами 42001, 42002 , 42003, 42004 ,42005, 42006, 42007, 42008, 42009, 42010;
- Транк в сторону шлюза.

12.6.2 Подготовка к работе

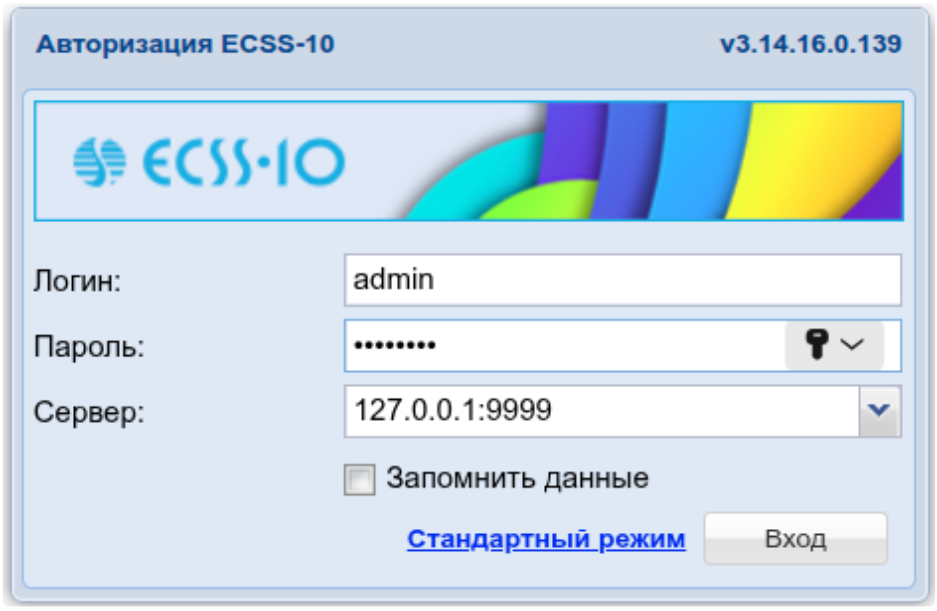


Рисунок 3 — Вход в web-конфигуратор (окно авторизации)

В окне авторизации необходимо ввести значения, определенные при установке web-конфигуратора.

> ⚠ Стандартные значения для авторизации:
> **Логин:** admin
> **Пароль:** password

После входа в систему будет видна основная рабочая область с иконками приложений, а также строка состояния с доступными опциями, в частности:

- 1 — выход из системы;
- 2 — выбор домена;
- 3 — выбор языка.

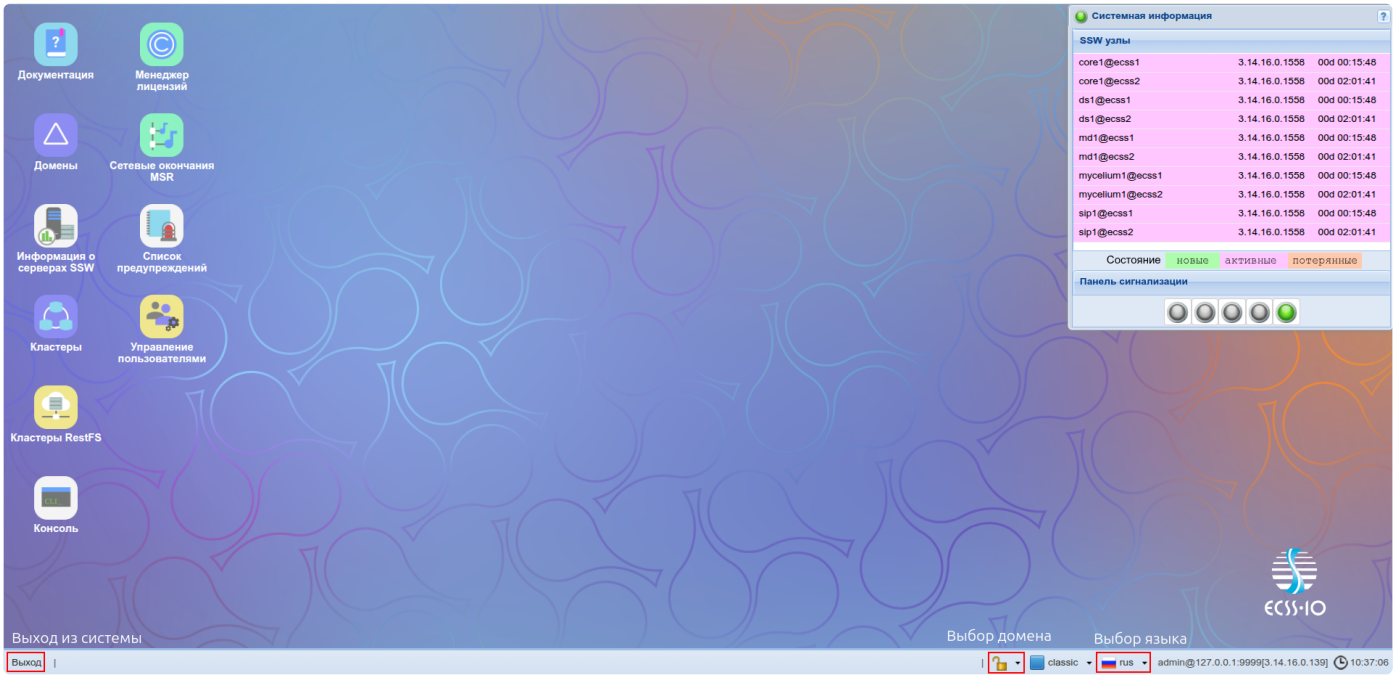


Рисунок 4 — Вид рабочей области web-конфигуратора

12.6.3 Создание учетной записи оператора

После авторизации для повышения безопасности при эксплуатации программного коммутатора рекомендуется создание учетных записей для операторов, а также сменить пароль для пользователя **admin**.

Для создания новой учетной записи оператора используется приложение "Управление пользователями" ("User manager"):

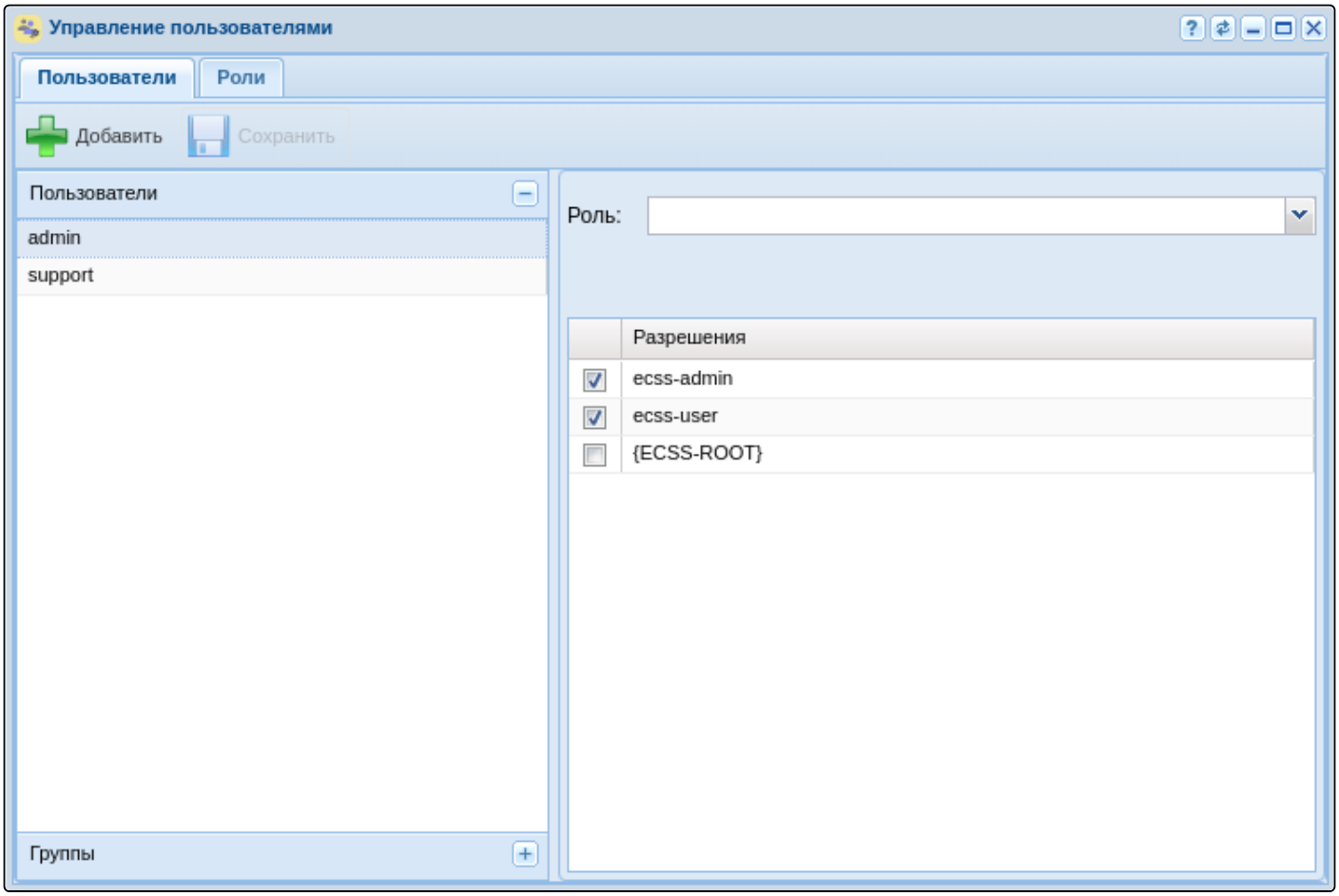


Рисунок 5 — Вид приложения "*Управление пользователями*"

Нажмите кнопку "*Добавить*"  . В открывшемся окне требуется определить новую учетную запись, для этого:

1. В поле "*Имя*" введите логин учетной записи, например "test";
2. В поле "*Пароль*" и "*Подтверждение*" введите пароль для пользователя, например "testpassword";
3. Определите уровень прав доступа для пользователя через выбор текущих разрешений или с помощью ролей, например **ecss-user**. Описание можно посмотреть в разделе "Управление пользователями".

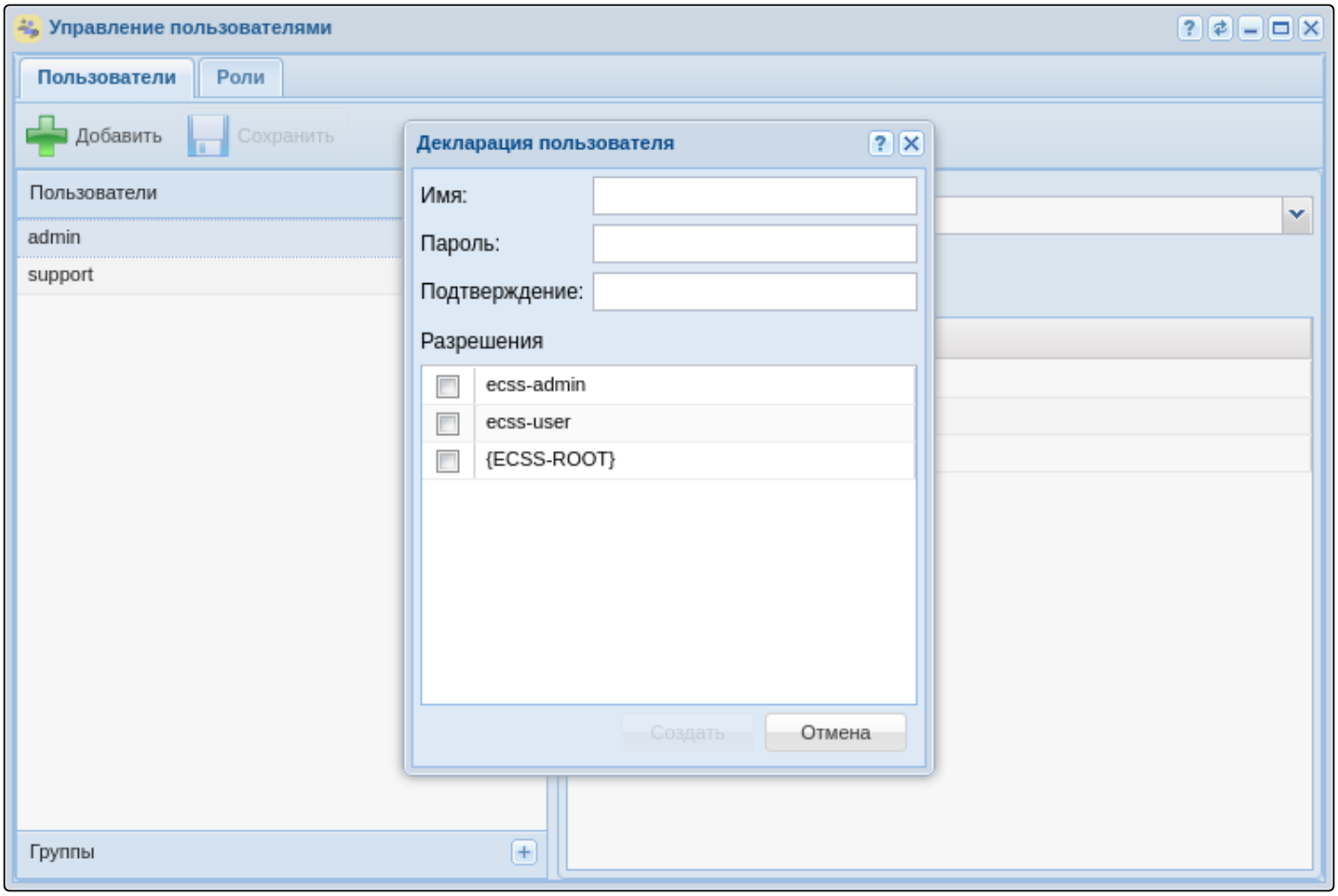


Рисунок 6 – Диалоговое окно создания учетной записи оператора

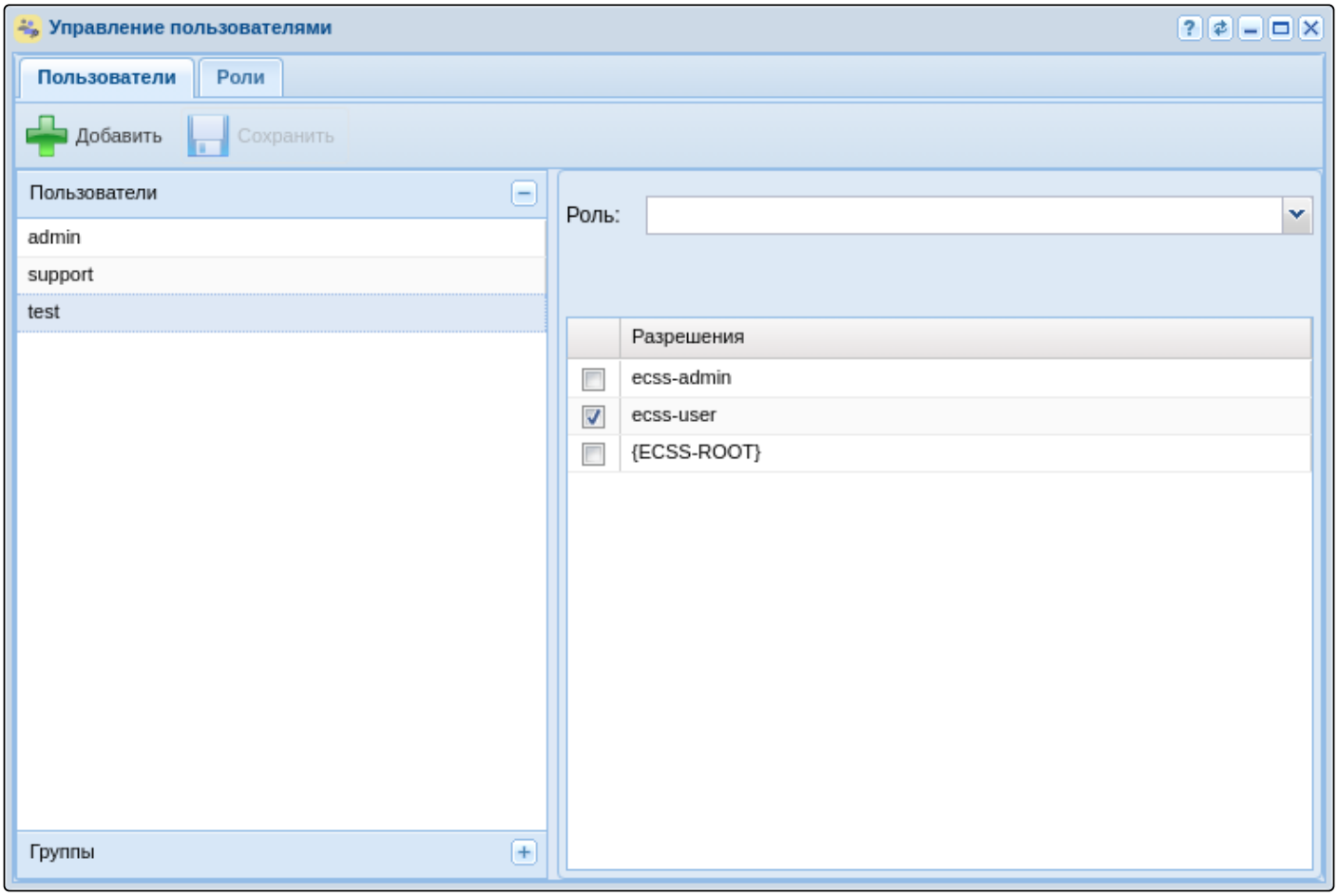


Рисунок 7 — Вид приложения с созданной учетной записью оператора

Для смены пароля рядом с названием пользователя нажмите на кнопку редактировать. В появившемся диалоговом окне введите:

1. Старый пароль (для пользователя "**admin**" по умолчанию паролем является "**password**");
2. Новый пароль;
3. Подтвердите новый пароль.

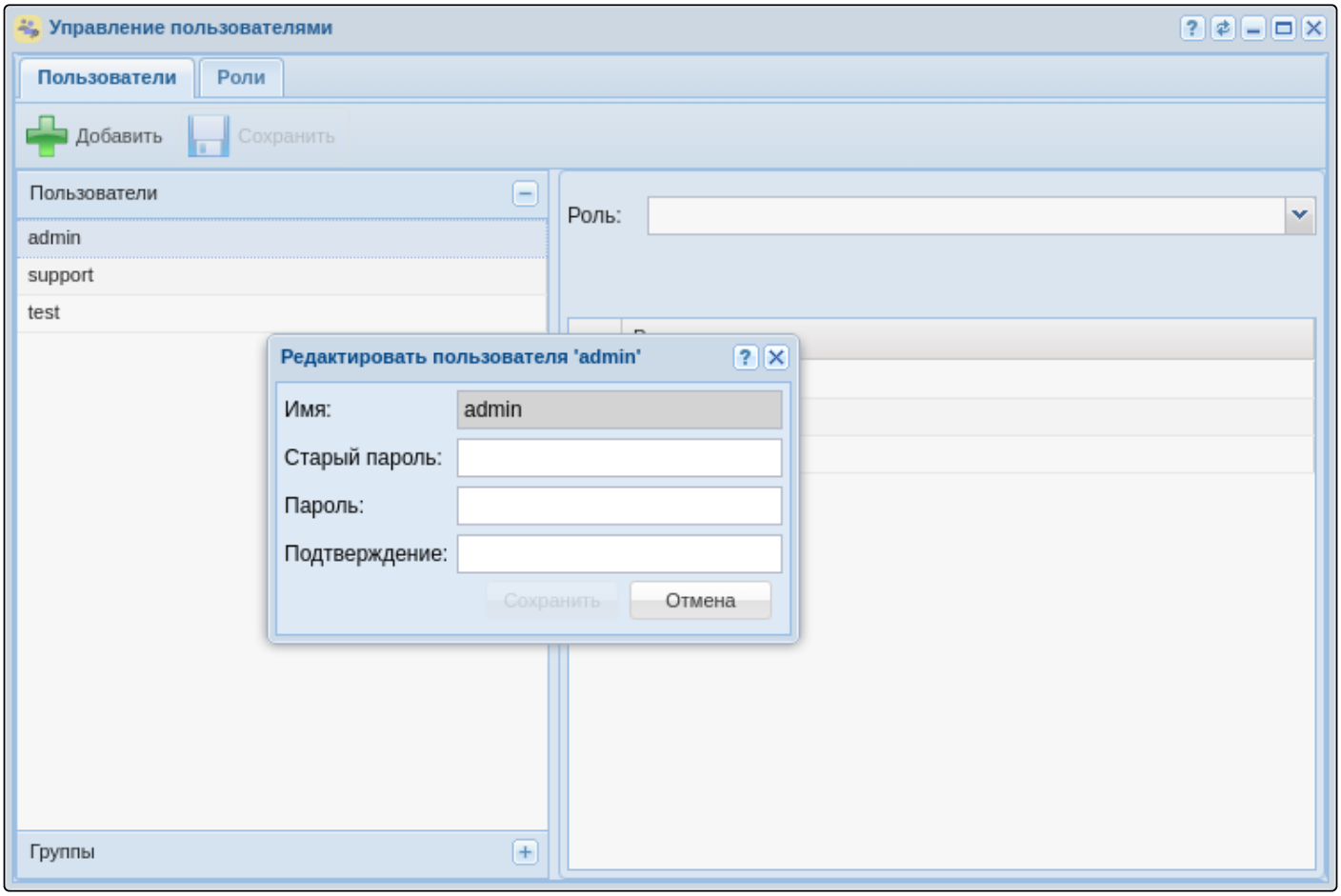
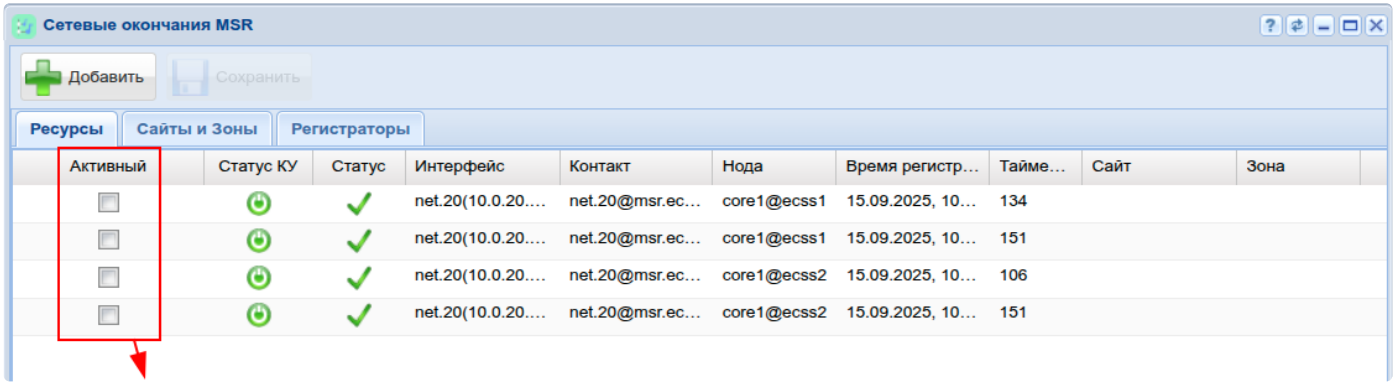


Рисунок 8 — Диалоговое окно редактирования учетной записи

12.6.4 Добавление медиа-ресурсов.

Для того чтобы начать работу с медиа-сервером сразу после его регистрации, необходимо активировать сетевые интерфейсы MSR. Приложение web-конфигуратора "Сетевые окончания MSR" ("MSR registrars") служит для настройки медиа-ресурсов.



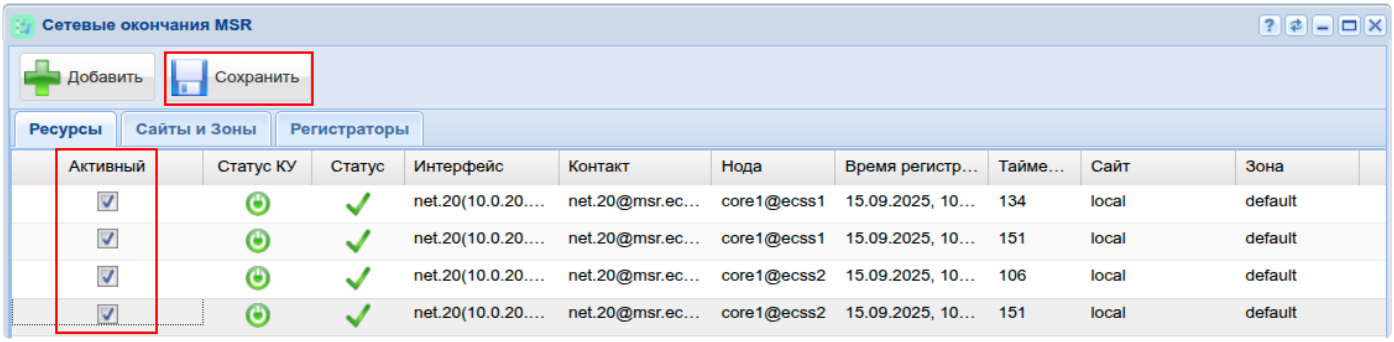
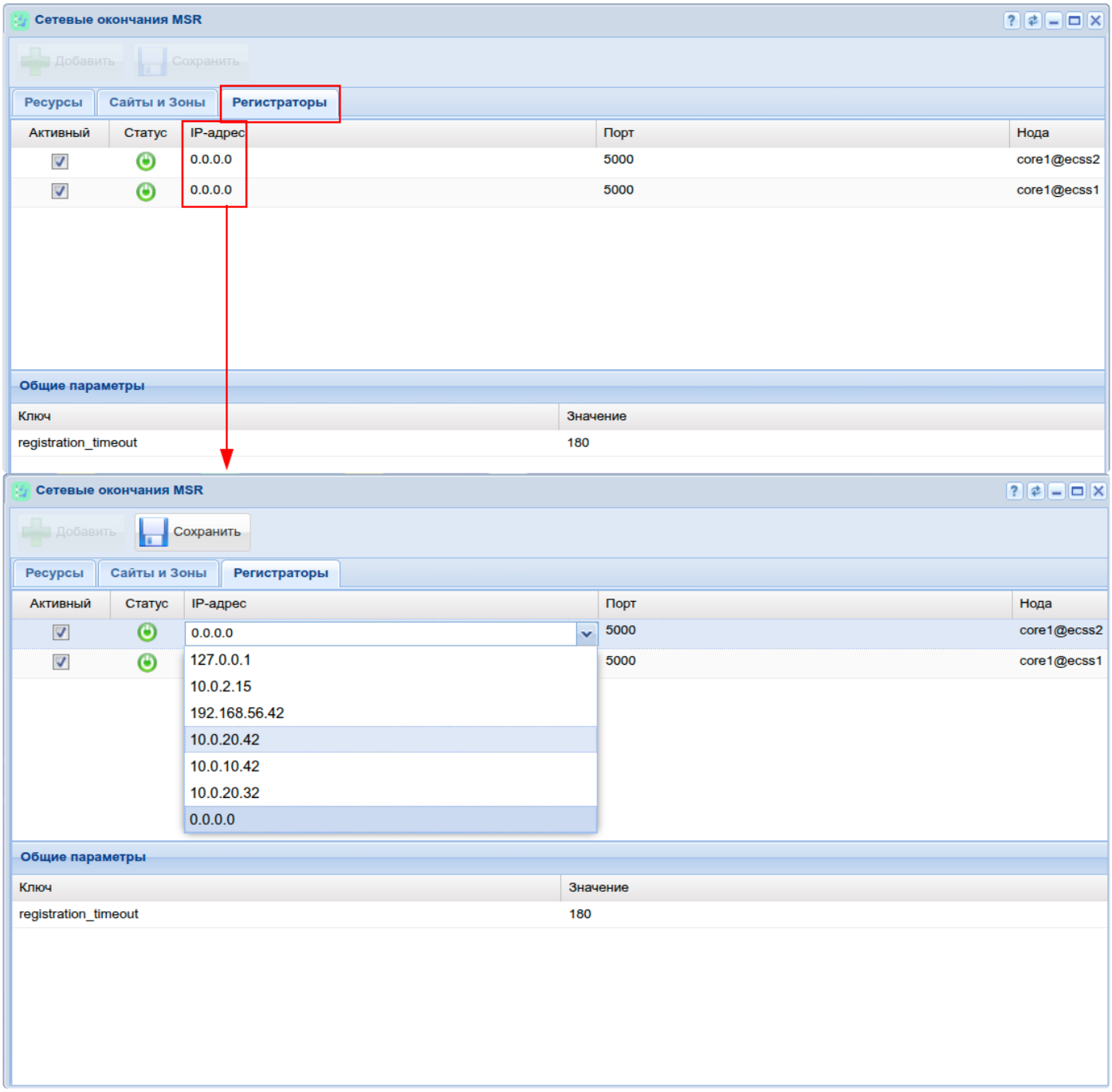


Рисунок 9 — Сетевые окончания MSR

Установить ipадрес для приема соединений от медиасервера, изменив значение 0.0.0.0 на реальные адреса MSR:



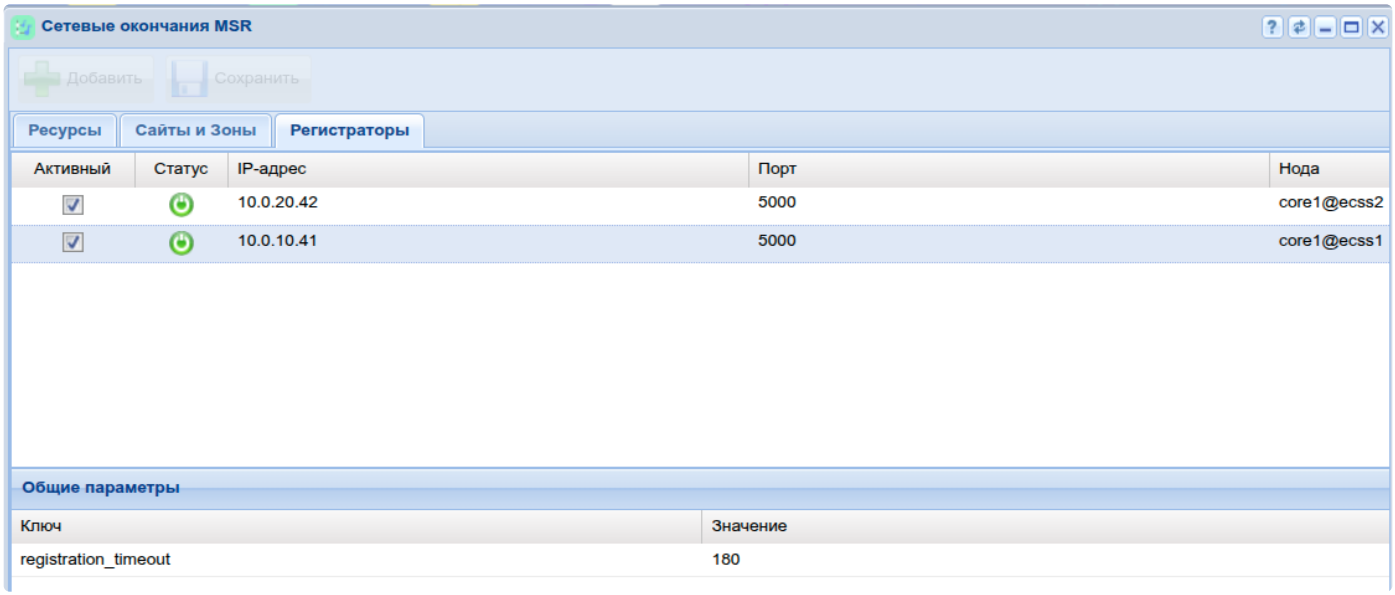


Рисунок 10 — Установка IP-add регистратора MSR

12.6.5 Создание IP-set (sip-транспорта) и присвоение его к домену

IP-set — совокупность одного или нескольких IP-адресов из одной подсети и списка слушающих UDP/TCP-портов, которые будут открыты SIP-адаптером на данных адресах.

Для настройки интерфейса необходимо открыть приложение "Кластеры ("Clusters").

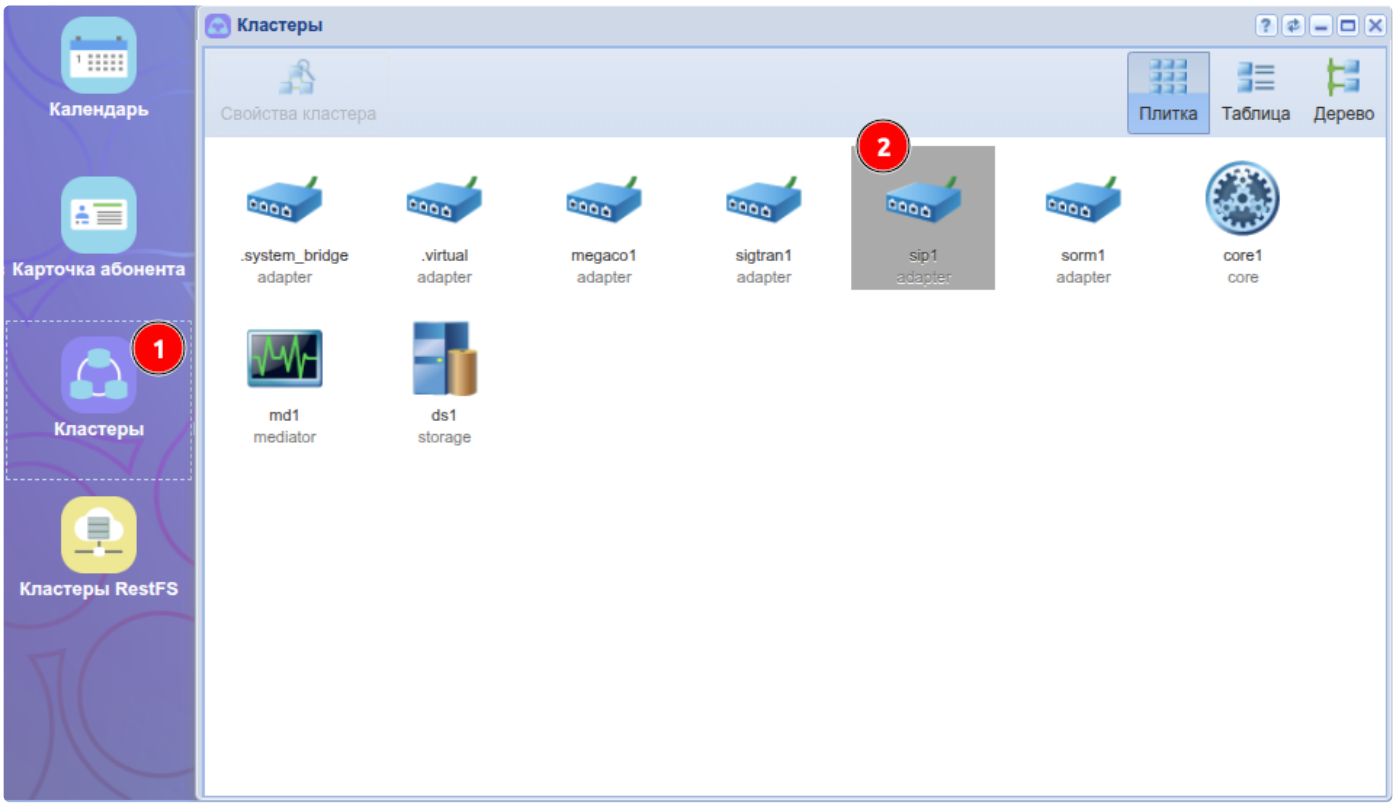


Рисунок 11 — Вид приложения "Кластеры"

Для создания новой группы IP-адресов (IP-set) выделите кластер SIP-адаптера "sip1" и нажмите на кнопку *"Свойства кластера"* (или нажмите на иконку кластера двойным щелчком левой кнопки мыши).

В появившемся диалоговом окне перейдите на вкладку *"Транспорт"*. Далее нажмите кнопку *"Добавить"*. Появится новая группа, для редактирования полей двойным щелчком кнопки мыши нажмите на нужное:

1. Переименуйте группу адресов (IP-set), например "test_set";

2. Задайте порт, по которому будет происходить обращение к домену, например 5060;

3. Разверните вновь созданную группу нажатием на треугольник слева от названия группы;

4. Определите адрес для ноды SIP-адаптера, согласно примеру конфигурации. Для системы без резервирования укажите 10.0.20.31:

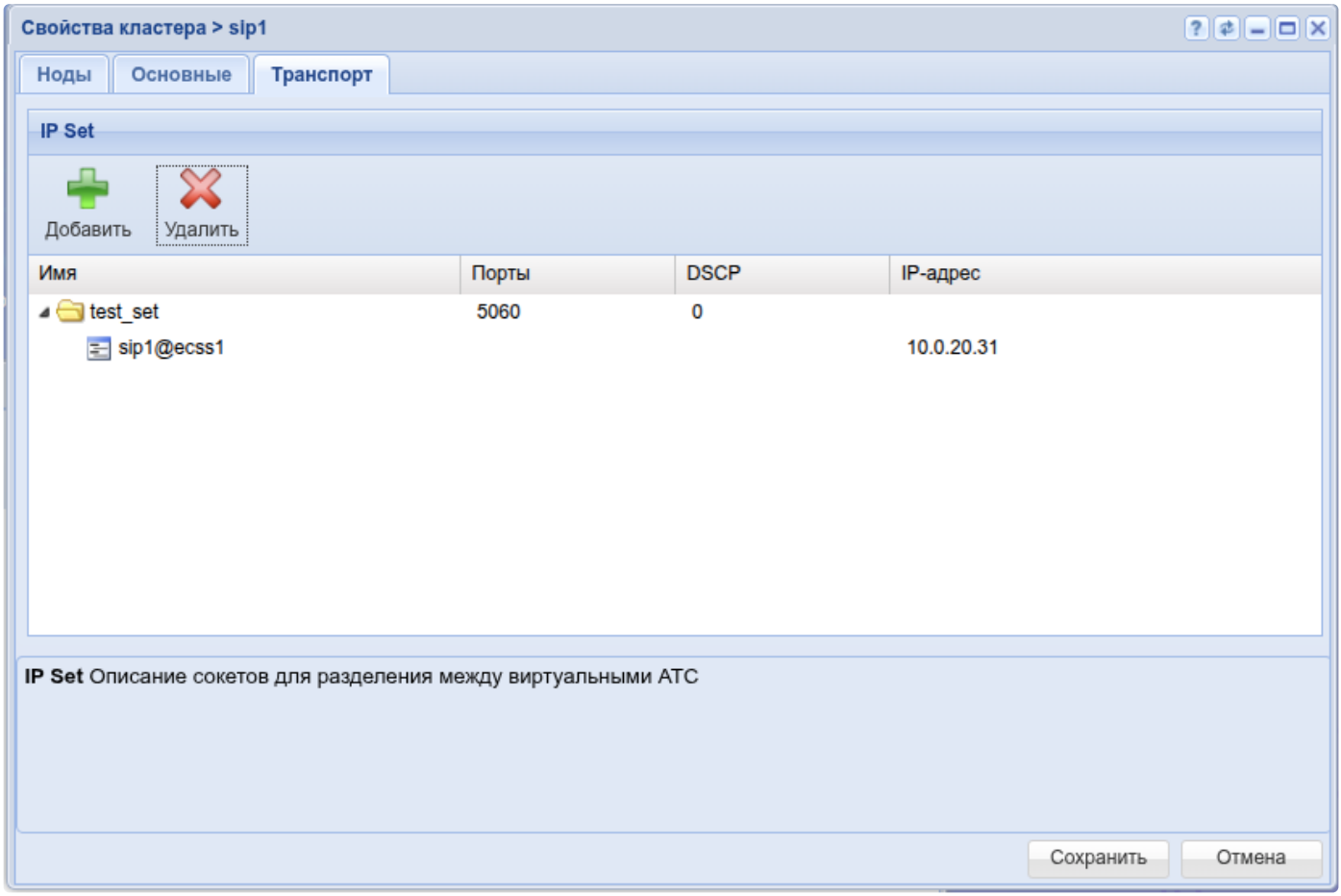


Рисунок 12 — Вид приложения "Кластеры/Транспорт"

Для системы с резервированием укажите 10.0.20.31 и 10.0.20.32:

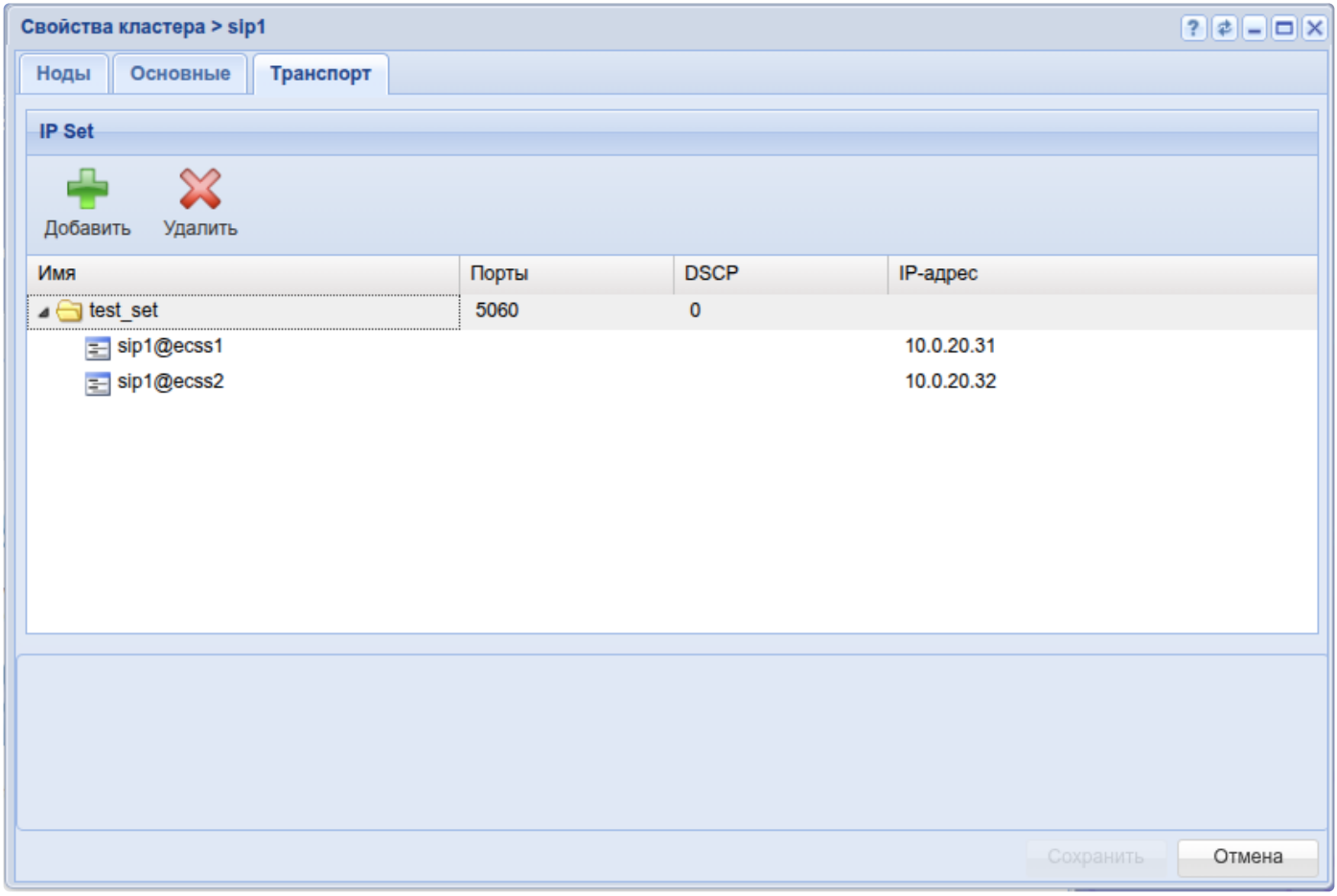


Рисунок 13 — Вид приложения "Кластеры/Транспорт" для кластера

Нажмите кнопку *"Сохранить"* для применения настроек.

12.6.6 Создание домена

Домен - Виртуальная PBX внутри нашего SSW. Таких виртуальных PBX может быть создано необходимое для проекта кол-во.

Для создания домена необходимо войти в приложение *"Домены" ("Domain")*. В открывшемся окне требуется создать домен, для этого:

1. Нажмите кнопку "*Добавить домен*":

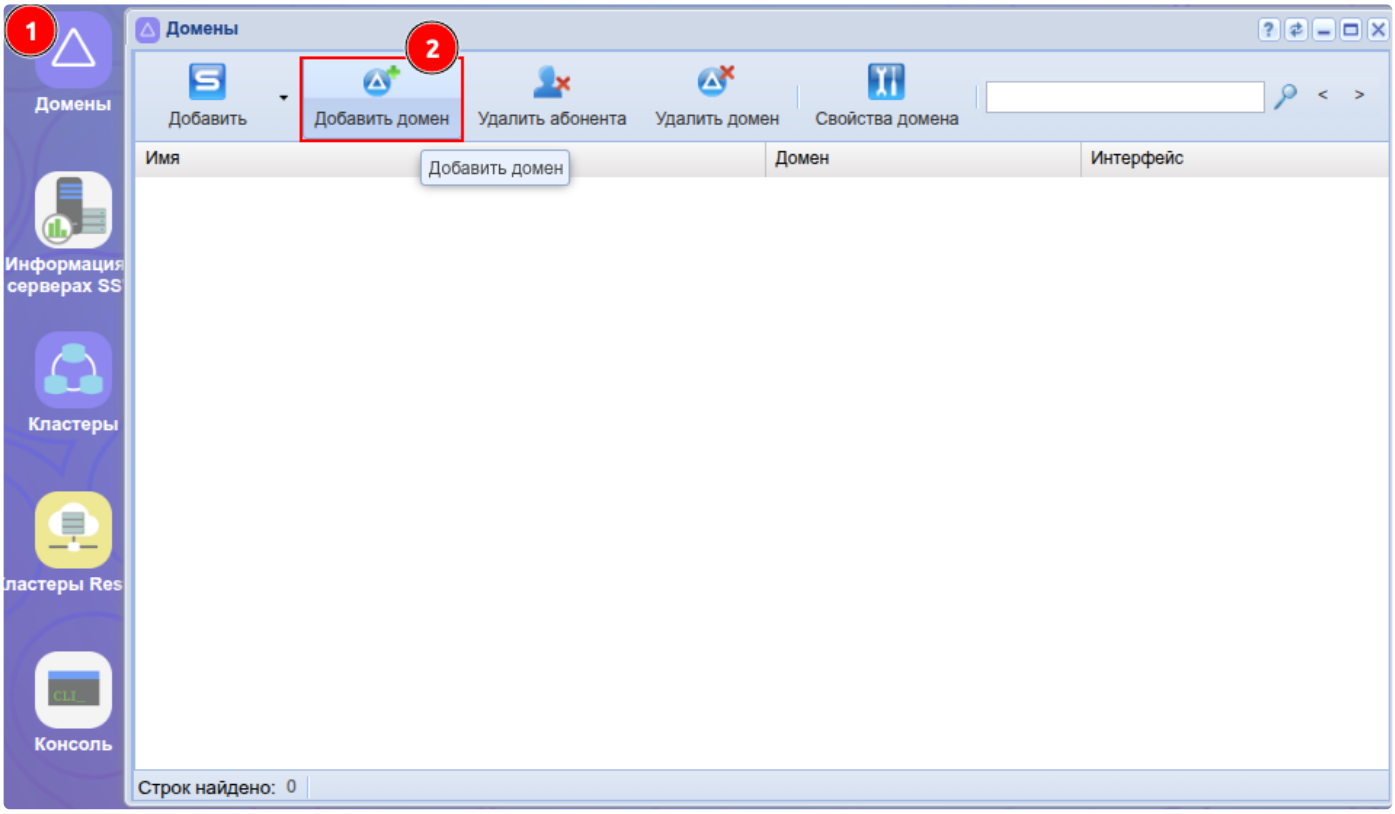


Рисунок 14 — Добавление домена в систему

2. В появившемся диалоговом окне доступны следующие настройки:

- *Имя (Name)* — индивидуальное имя виртуальной АТС;
- *Профиль услуг (SS profile)* — системный профиль дополнительных услуг. Этот профиль будет скопирован с тем же именем во вновь создаваемый домен и для всех услуг из этого профиля будет автоматически разрешён доступ через access-list;
- *IVR профиль (IVR profile)* — профиль IVR, заданный в приложении "Редактор IVR ограничений".

Введите название домена, например "test_domain";

3. Нажмите кнопку "*Ок*":

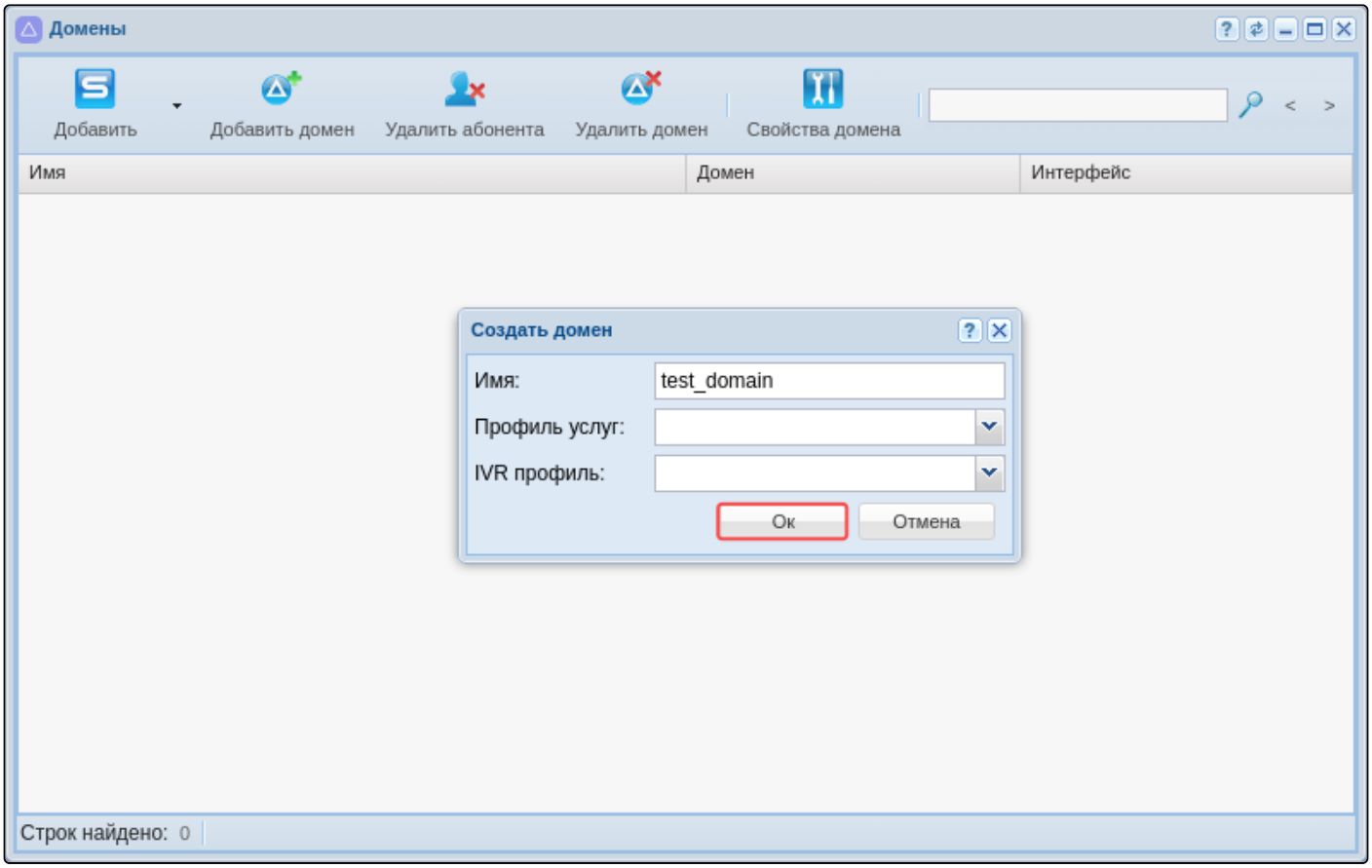


Рисунок 15 – Определение настроек нового домена

4. Нажмите кнопку *"Обновить"* в приложении.

Созданный домен будет отображаться в текущей конфигурации:

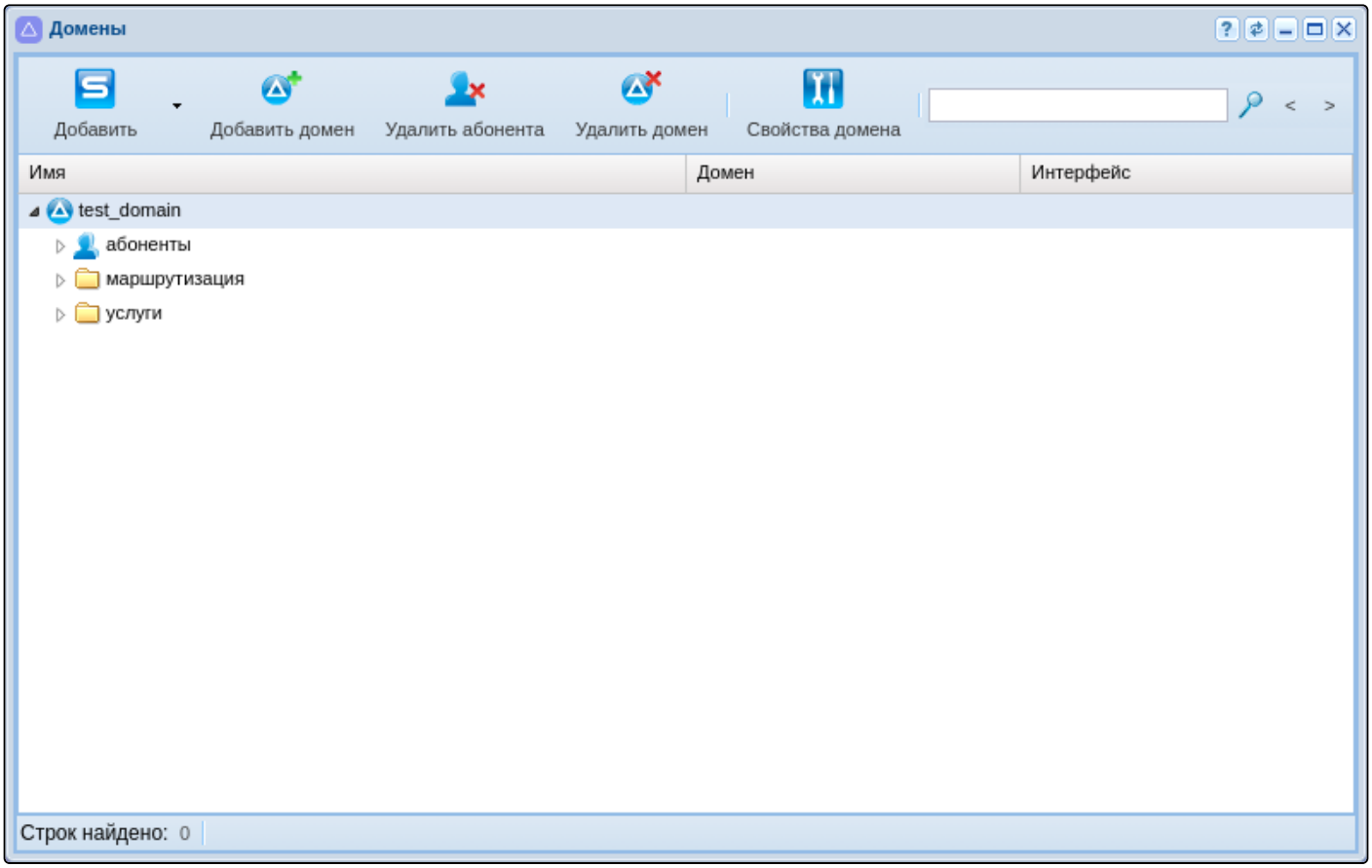


Рисунок 16 — Отображение созданного домена

Для редактирования текущего домена его необходимо выбрать в системе. Для перехода в домен воспользуйтесь опцией выбора домена (см. пункт 2 на рисунке "Вид рабочей области web-конфигуратора").

После выбора домена, согласно текущей конфигурации системы, станут доступны все приложения:

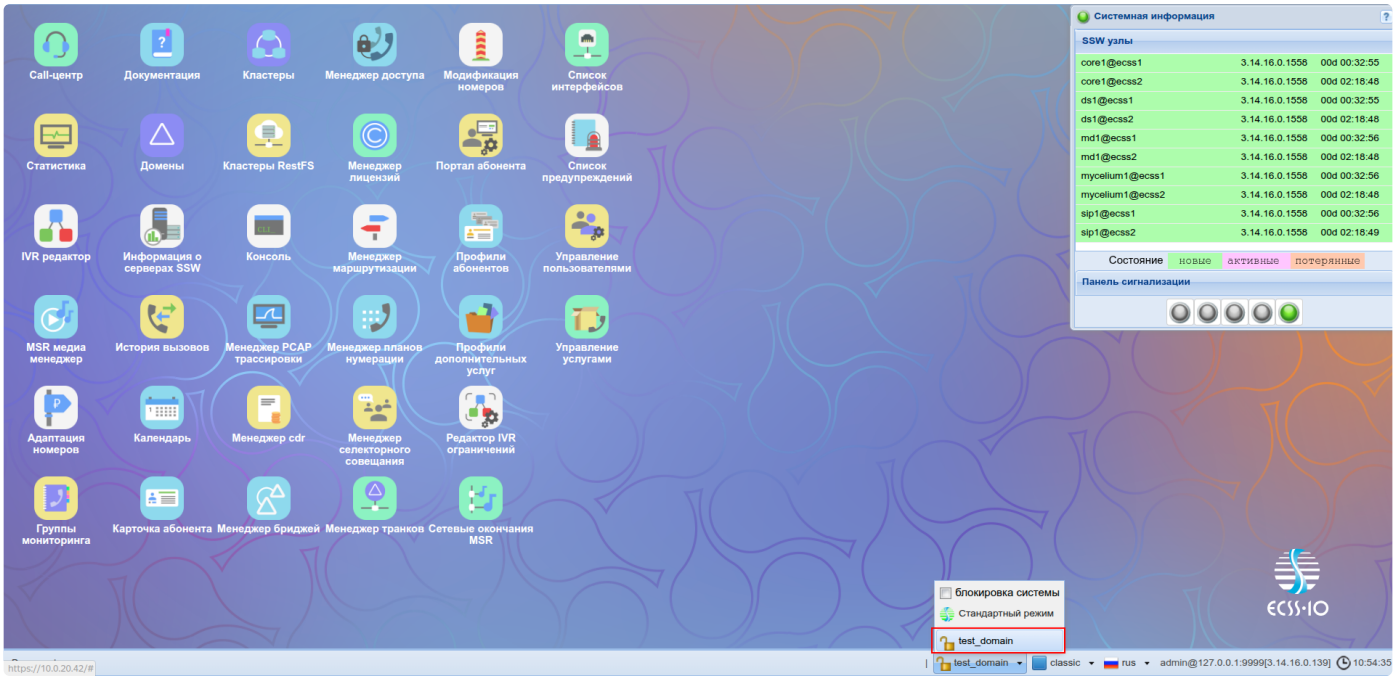


Рисунок 17 — Отображение приложений в текущей конфигурации системы

Для того чтобы **привязать группу адресов к домену**, необходимо вернуться в приложение *"Домены" ("Domain")*, выделить домен и перейти в настройки нажатием на кнопку *"Свойства домена"* или двойным нажатием левой кнопки мыши на домен.

В настройках списка откройте ветку "SIP", в ней "SIP транспорт", далее выберите в поле IP set созданную группу адресов. Нажмите кнопку *"Сохранить"* для применения настроек.

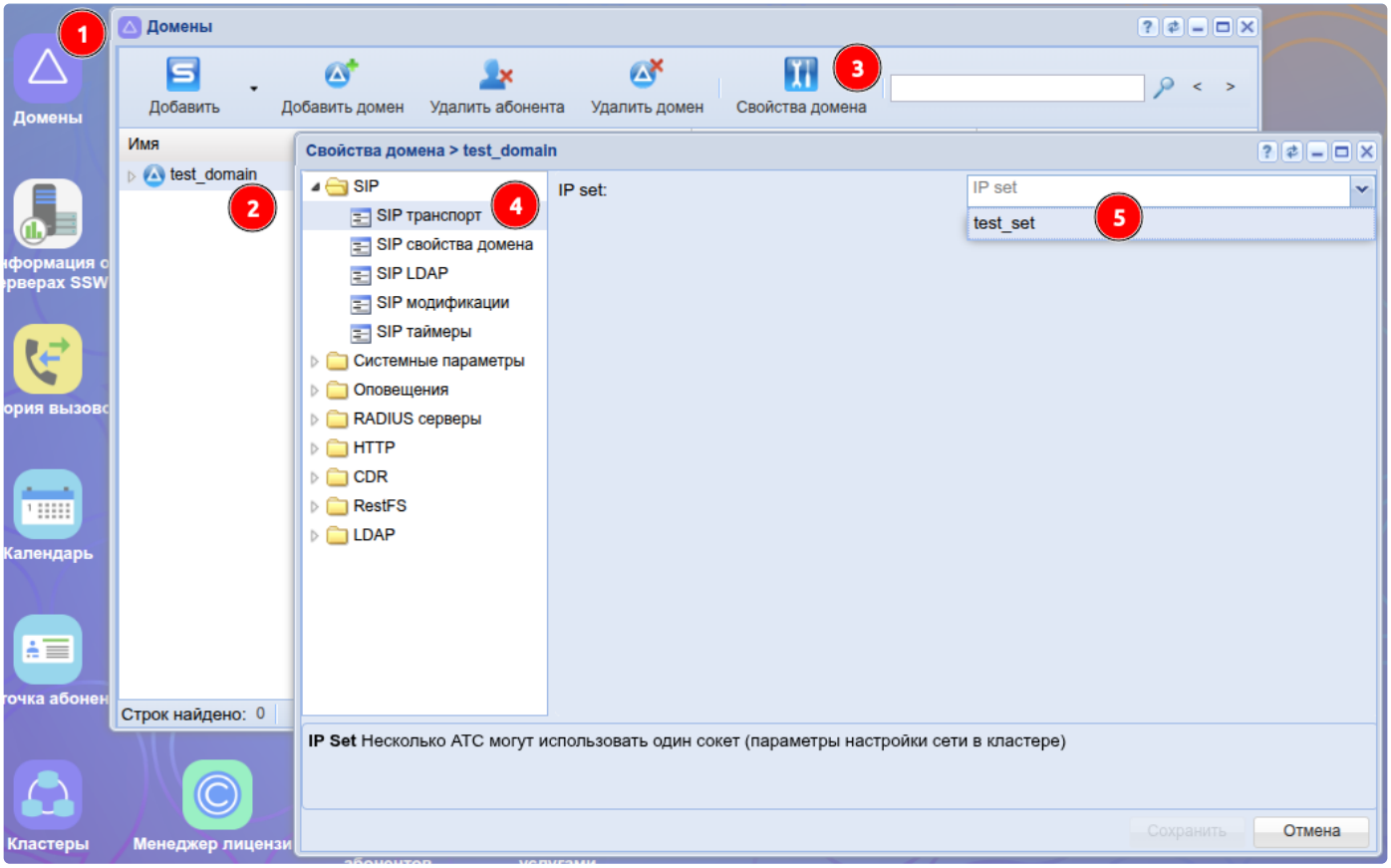


Рисунок 18 — Окно настройки для SIP транспорта

Выбрать требуемые услуги для созданного домена, которые будут использоваться (для примера для абонентского домена транковые услуги не востребованы , и наоборот для транзитного домена не нужно указывать услуги для абонента).

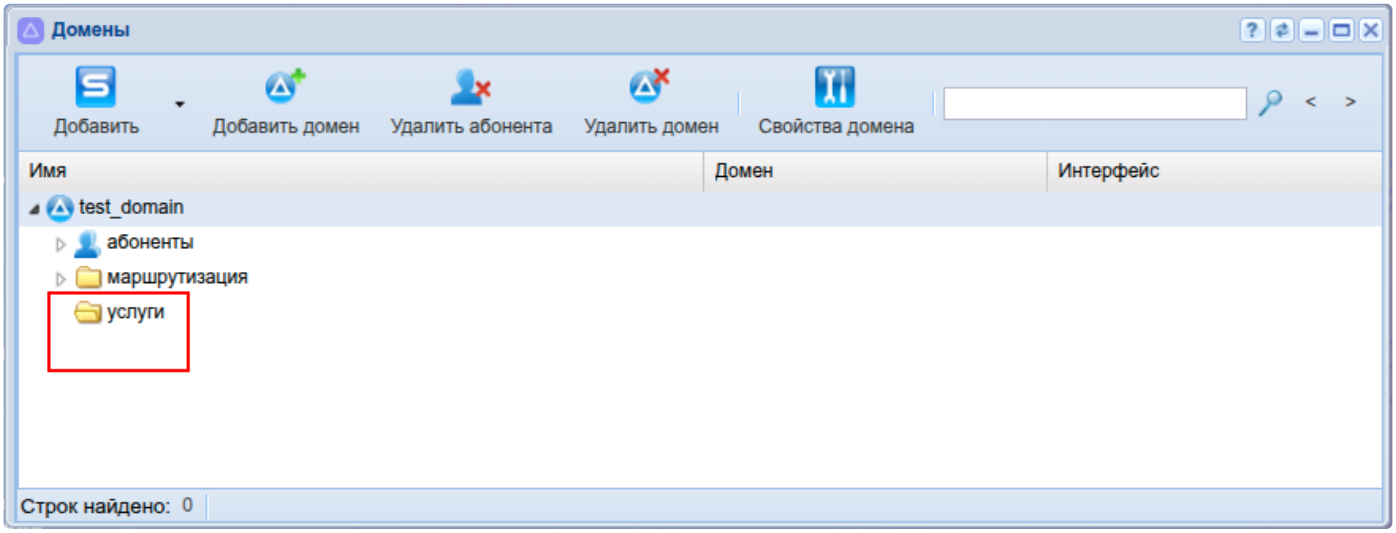


Рисунок 19 — После создания нового домена, услуги отсутствуют.

1. Выбрать приложение "Управление услугами"
2. Выбрать колонку "Установленные услуги" (услуги указанные в лицензии для SSW)

3. Выбрать необходимые для указанного домена услуги
4. Сохранить конфигурацию

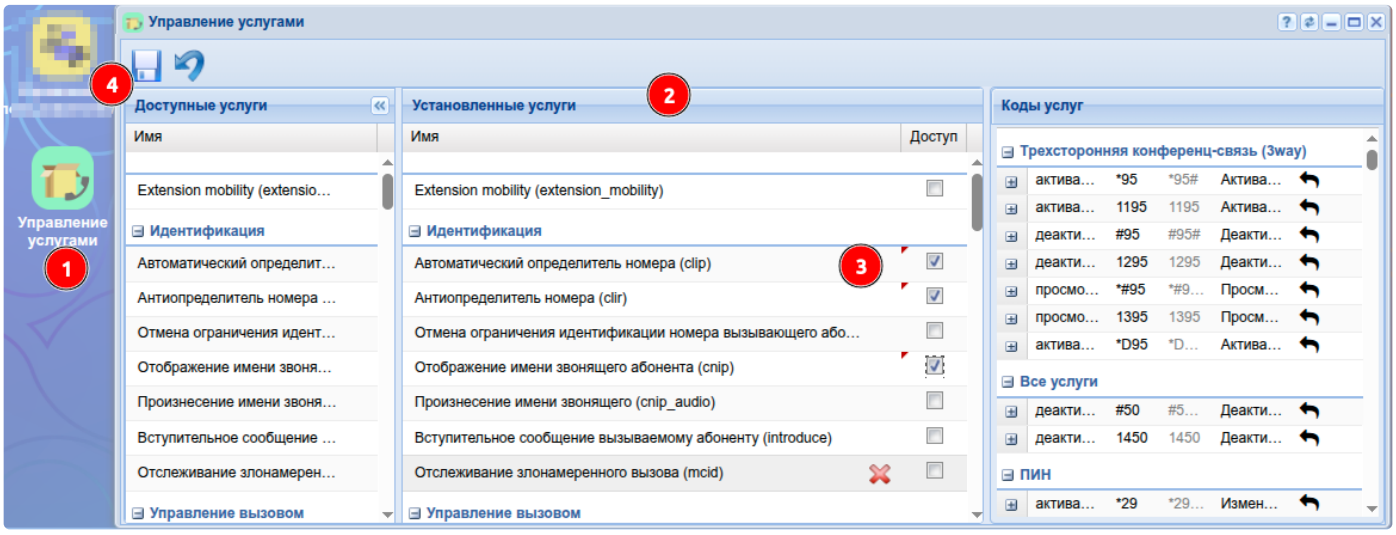


Рисунок 20 — Выбор услуг для абонентов / транков нового домена.

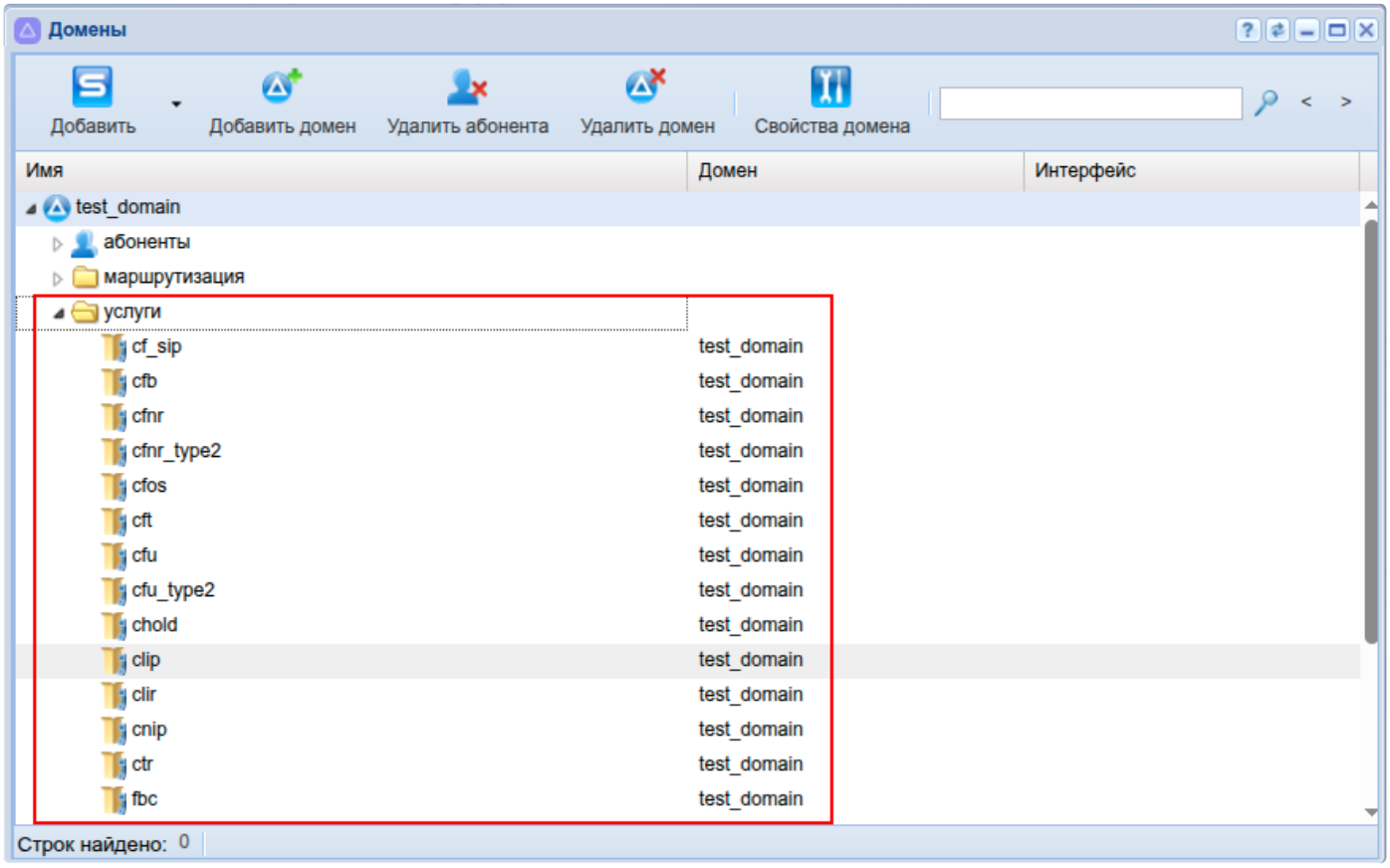


Рисунок 21 — Список услуг теперь доступных для абонентов нового домена.

12.6.7 Создание абонентов

Для создания и редактирования параметров абонентов в системе используется приложение "Карточка абонента" ("Subscriber card").

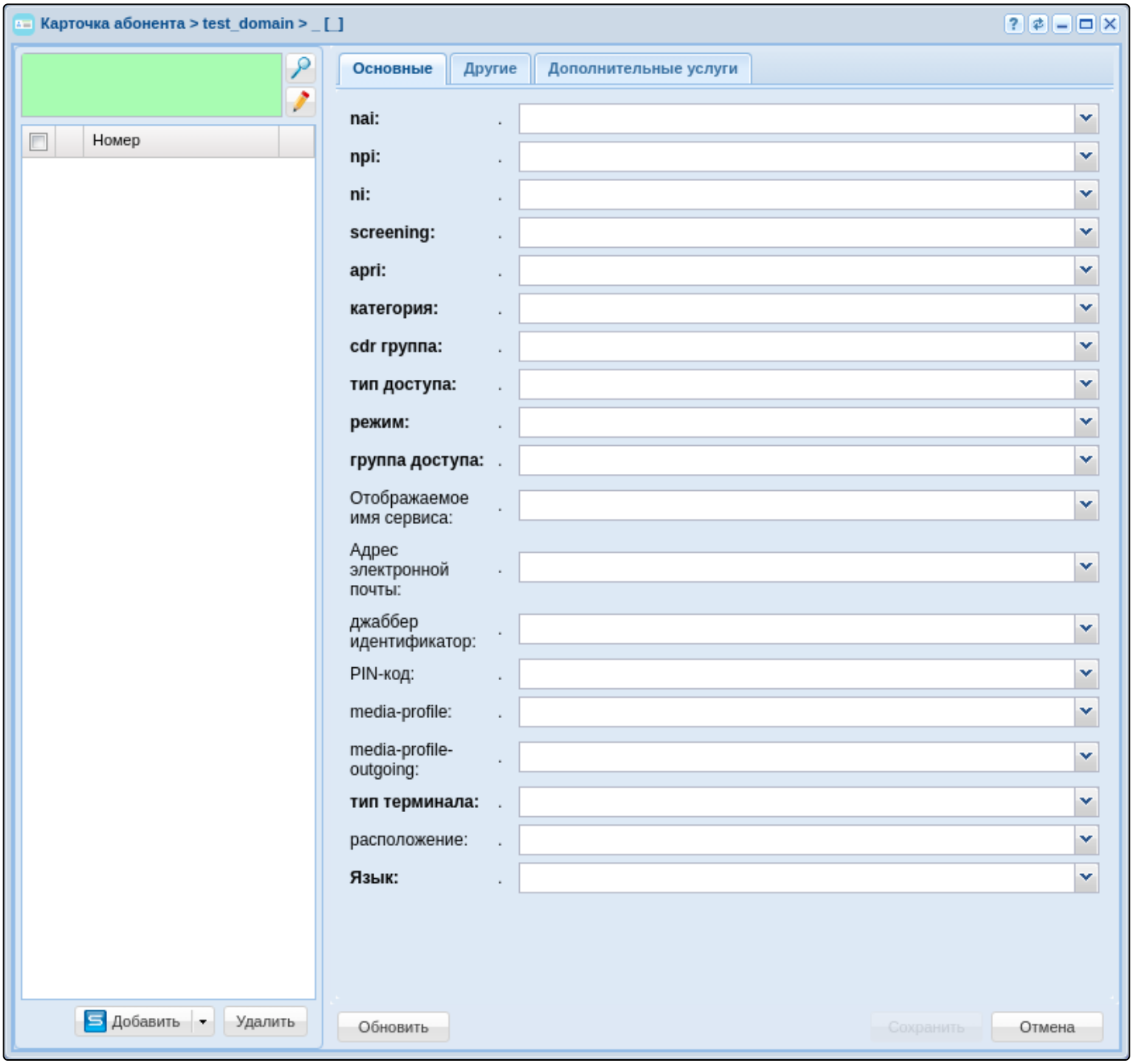


Рисунок 22 — Вид приложения "Карточка абонента"

В конфигурации возможно создание SIP-абонентов и виртуальных абонентов.

Для пользователей, имеющих физическое окончание, используется функционал SIP-абонента, в то время как виртуальный абонент используется тогда, когда нужна функциональность без физических окончаний. Например, номер для обращения к ivr-скрипту.

Для создания новых абонентов нужно нажать на кнопку *"Добавить"*.

В появившемся диалоговом окне укажите следующие параметры:

- *Имя интерфейса* — номер или группа номеров, который присваивается абоненту, строка, например {42001-42010} или abcd;
- *Контекст* — контекст маршрутизации, выбираем тот, который создали, например "test_name", или по умолчанию - "default_routing";
- Группа интерфейсов — имя группы объединяющей нескольких абонентов, для примера "sip";
- *Алиас как пользователь* — настройка, которая связывает номер сущности, алиас и пользователя одним и тем же названием, в примере настройка активирована;

- *Авторизация* — порядок проверки подлинности прав пользователя на доступ к данным, в примере используется"register";
- *Логин* — использую настройку номер как логин для простоты;
- *Пароль* — можно задать или использовать сгенерированный системой.

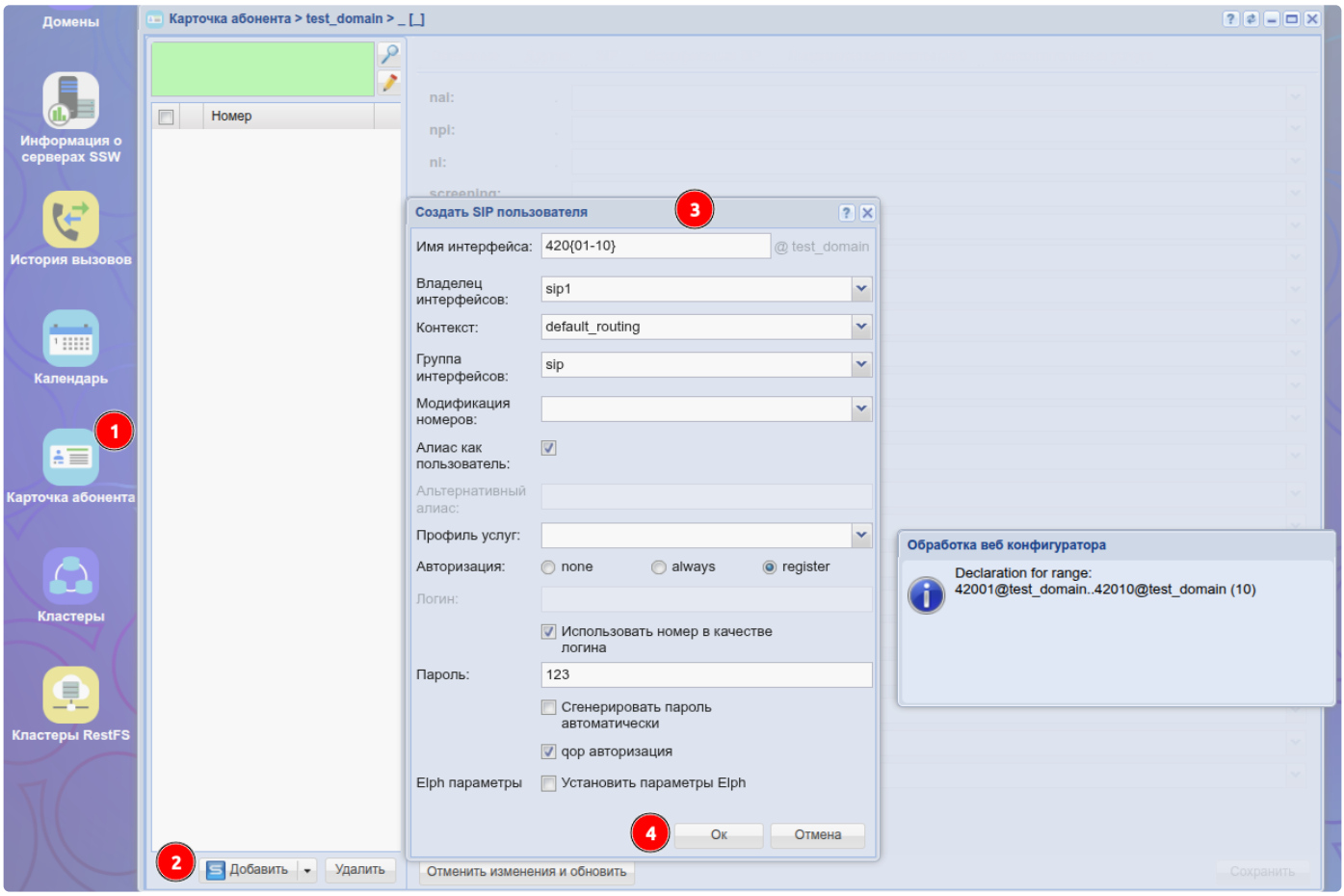


Рисунок 23 — Пример создания 10 абонентов в домене "test_domain"

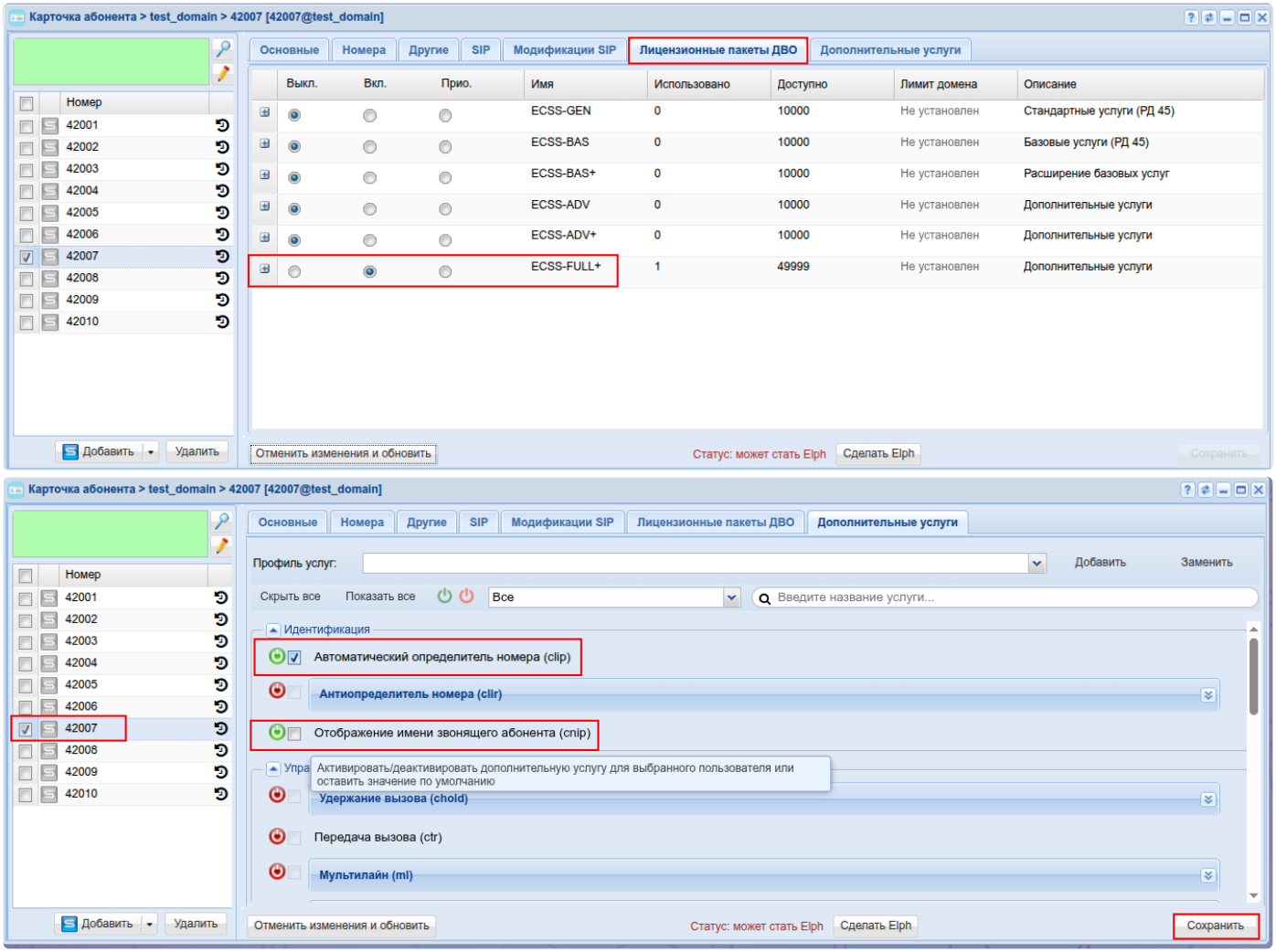


Рисунок 24 – Пример добавления услуг для абонента

12.6.8 Создание и применение контекстов маршрутизации для домена

Маршрутизация отвечает за поиск номера и последующую адресацию вызова. Для корректной работы системы должен быть настроен как минимум один контекст маршрутизации.

Настройка маршрутизации осуществляется в приложении "Менеджер маршрутизации" ("Routing manager") .

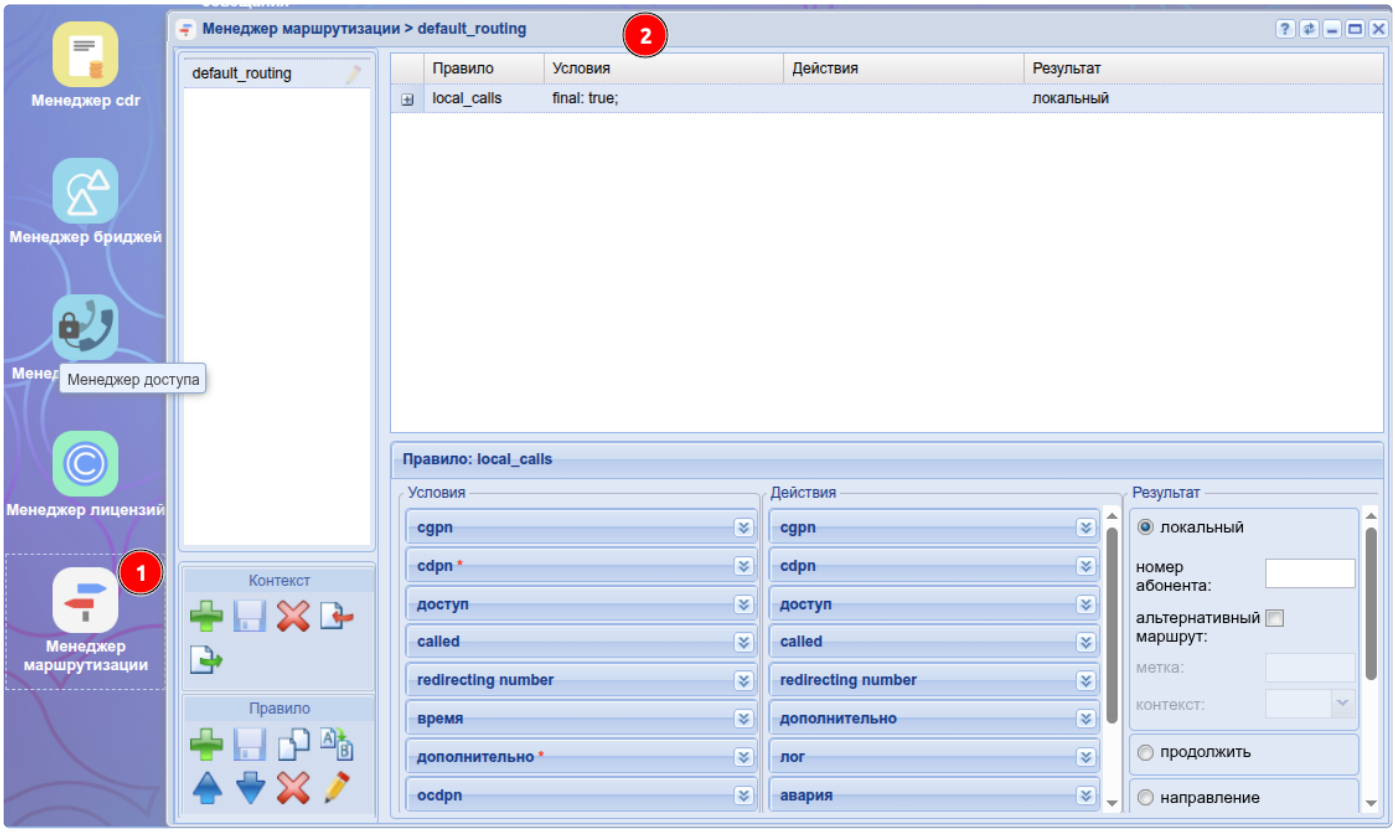


Рисунок 25 — Вид приложения "Менеджер маршрутизации"

В качестве примера создадим контекст, а затем несколько правил в нём, для этого:

1. В левой части окна в разделе "Контекст" нажмите на кнопку *"Создать контекст"* ;
2. В открывшемся диалоговом окне обозначьте имя контекста:

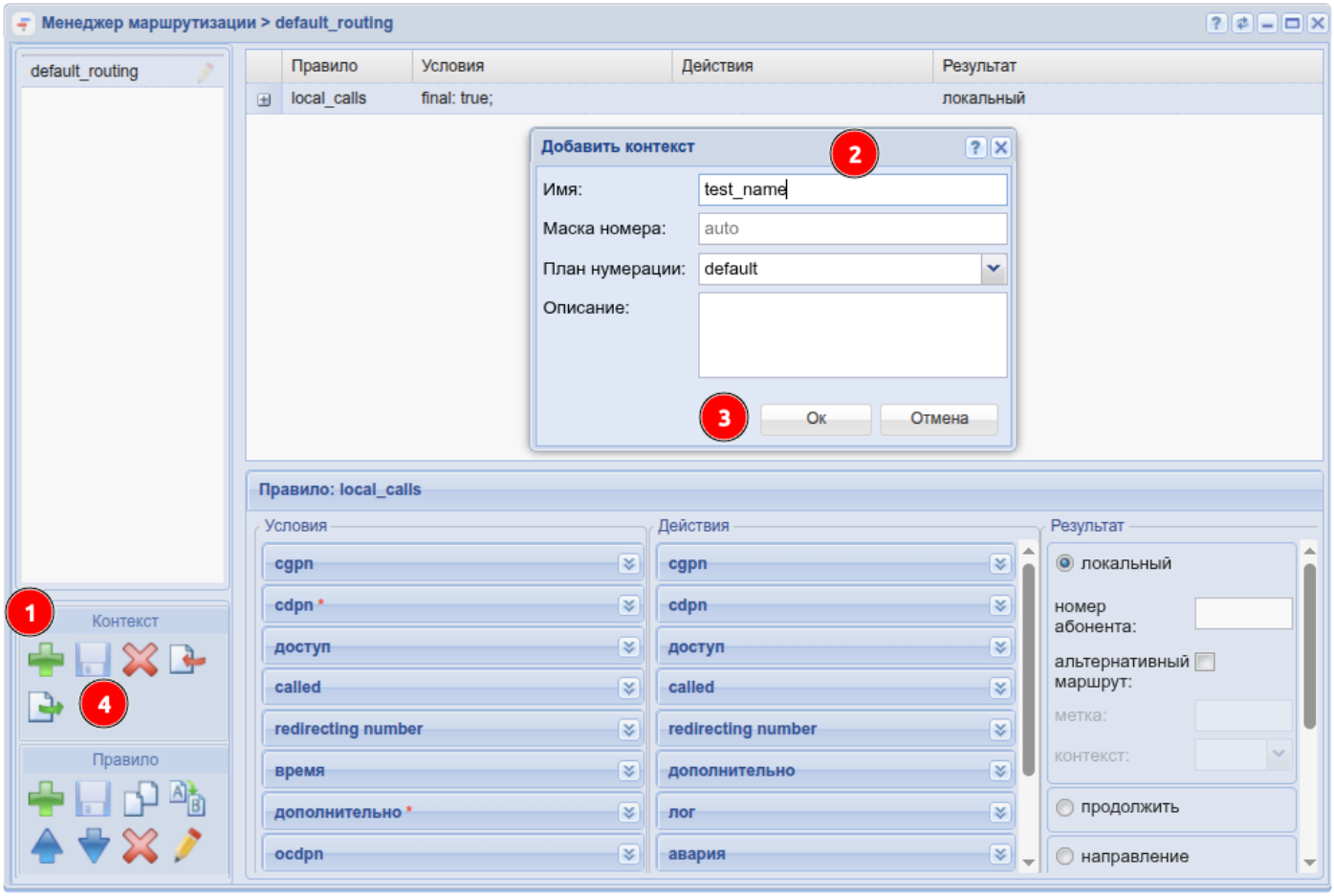


Рисунок 26 — Вид приложения "Менеджер маршрутизации" добавление нового контекста маршрутизации

3. Нажмите *"Сохранить контекст"*  ;

Создадим 4 правила в данном контексте:

- rule1 — правило для выхода на транк TAU-72;
- rule2 — правило локальной маршрутизации для номеров 42001-42005;
- rule3 — правило для выхода в ivr;
- rule4 — правило-исключение.

Для создания нового правила нужно выделить созданный контекст и нажать кнопку *"Создать правило"*. В появившемся окне нужно ввести название правила. Затем сохранить вновь созданные правила.

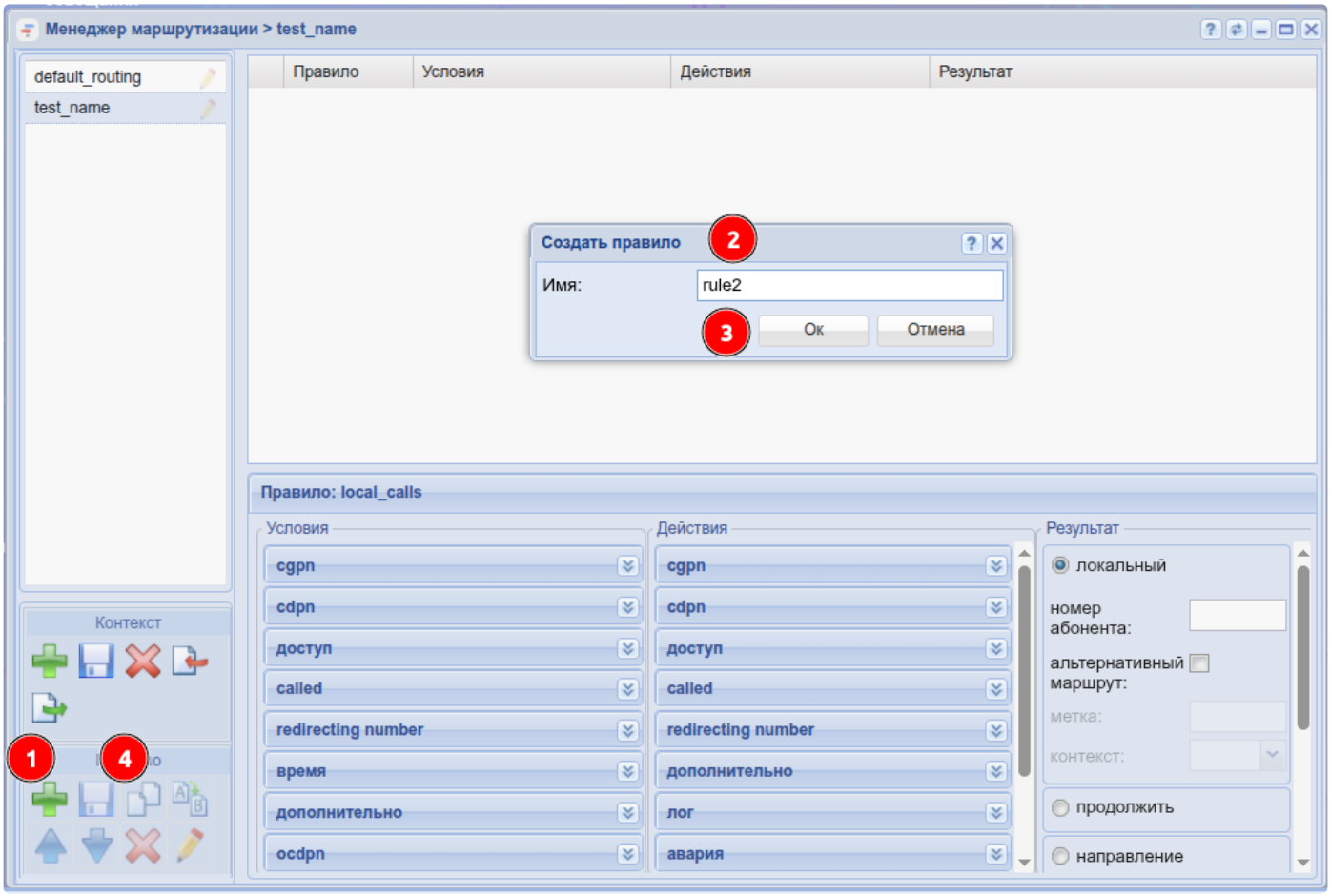


Рисунок 27 – Создание правила контекста маршрутизации

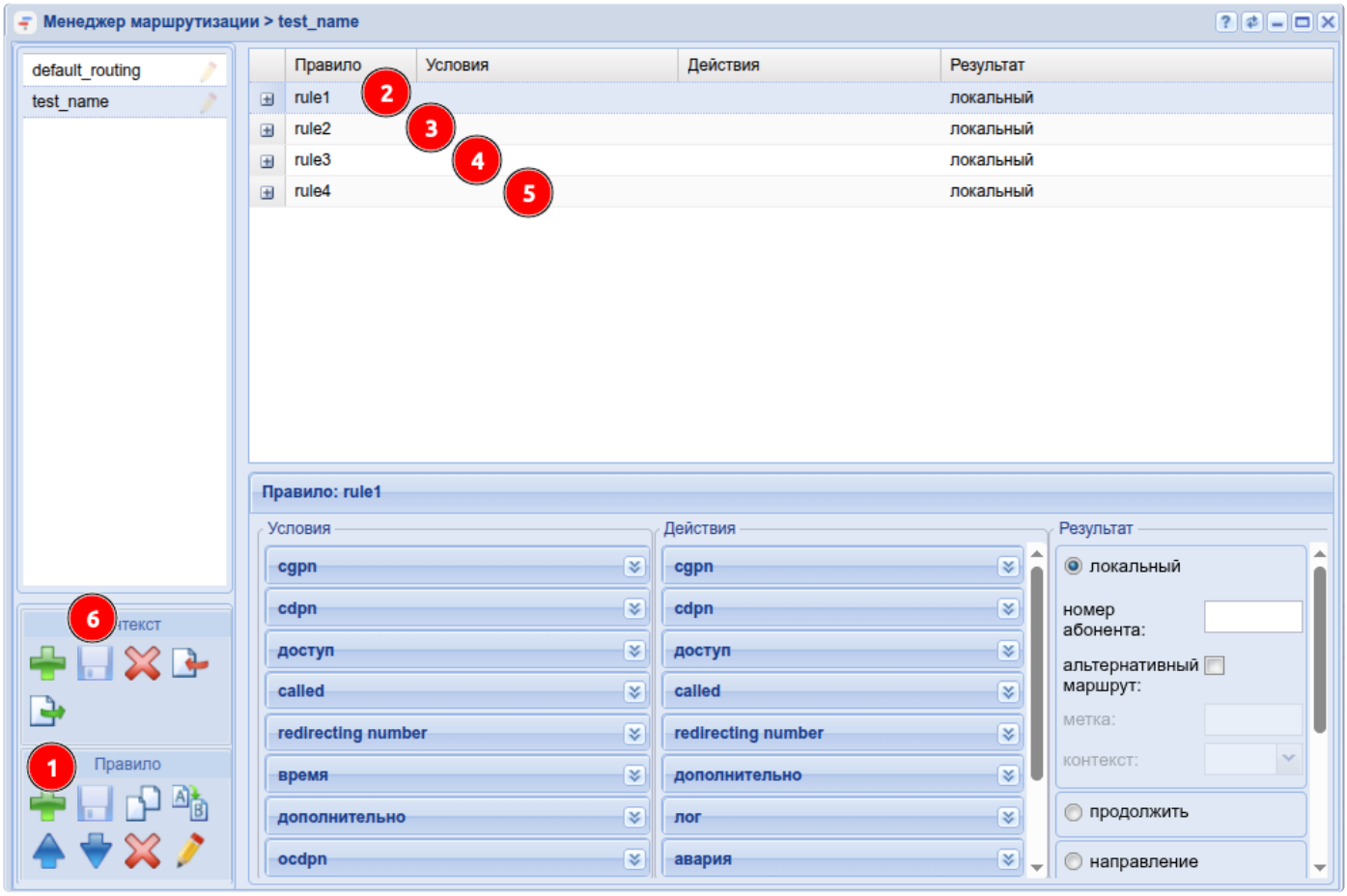


Рисунок 28 — Определение правил

На данный момент не определен транк, чтобы можно было сослаться на него в правиле, однако можно задать номера, по которым будет производится выборка.

Нажав на правило rule1, перейдем в нижнюю часть экрана, в которой располагаются области для редактирования контекста маршрутизации. Условно определим, что выборка для выхода на транк будет проводиться по признакам номера вызываемого абонента (CDPN), а номера в транке должны начинаться с цифры 4.

Функционально контекст маршрутизации делится на три части:

- Условия ("Condition") — раздел, определяющий выражения для выборки по предложенным признакам;
- Действия ("Action") — раздел, выполняющий преобразование признаков номеров к конкретному значению;
- Результат ("Result") — раздел, завершающий маршрутизацию и определяющий её результат.

rule1: Для выхода на транк нужно правильно отредактировать каждую часть:

- В разделе условий перейти на вкладку CDPN. Ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера с 42006 по 42007, нужно ввести условие: 420(6-7);
- В разделе действий перейти на вкладку CDPN. Ввести маску для изменения номера в поле "Номер". Например, чтобы добавить число 4 перед номером, нужно заполнить поле следующим выражением: 4{1,2,3,4,5};
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить транк в системе, поэтому к настройке данного правила вернемся чуть позже.

Таким же образом нужно настроить оставшиеся правила.

Для rule2:

- В разделе условий, во вкладке CDPN ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера со 42001 по 42005, ввести условие 420(1-5);
- В разделе результата определить результат как локальный (т.е. локальная маршрутизация);
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Для rule 3 допустим, что абоненты с номерами 108, 109 и 110 попадают в ivr-скрипт–информатор, прежде чем звонить далее.

- В разделе условий, во вкладке CGPN ввести номера телефонов, по которым будет выборка в поле "Номер", например 420(10,08-09).
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить в системе ivr-скрипт, поэтому к настройке данного правила вернемся чуть позже.

Для rule 4 мы определяем правило-исключение — это правило, которое работает в случае, если любые другие правила не срабатывают.

- По умолчанию данное правило создается в системе в контексте default_routing. В условиях на вызываемый номер прописывается символ %, а результатом является локальная маршрутизация, однако, если мы назначаем абоненту контекст, отличный от default_routing, в конце *рекомендуется* создавать это правило.

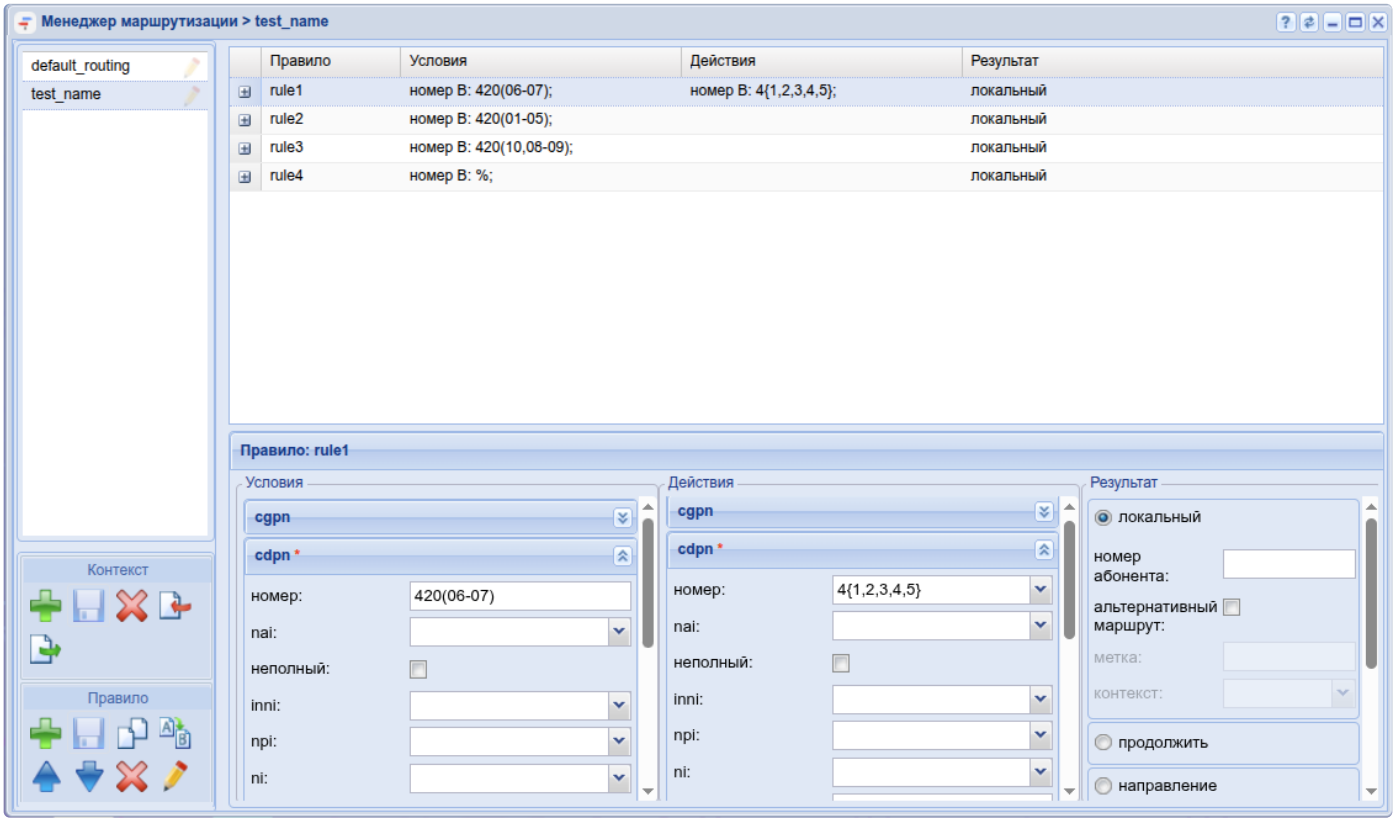


Рисунок 29 — Пример настроек контекста маршрутизации

12.6.9 Создание транка

Для создания и редактирования параметров транка в системе используется приложение "Менеджер транков" ("Trunk manager") .

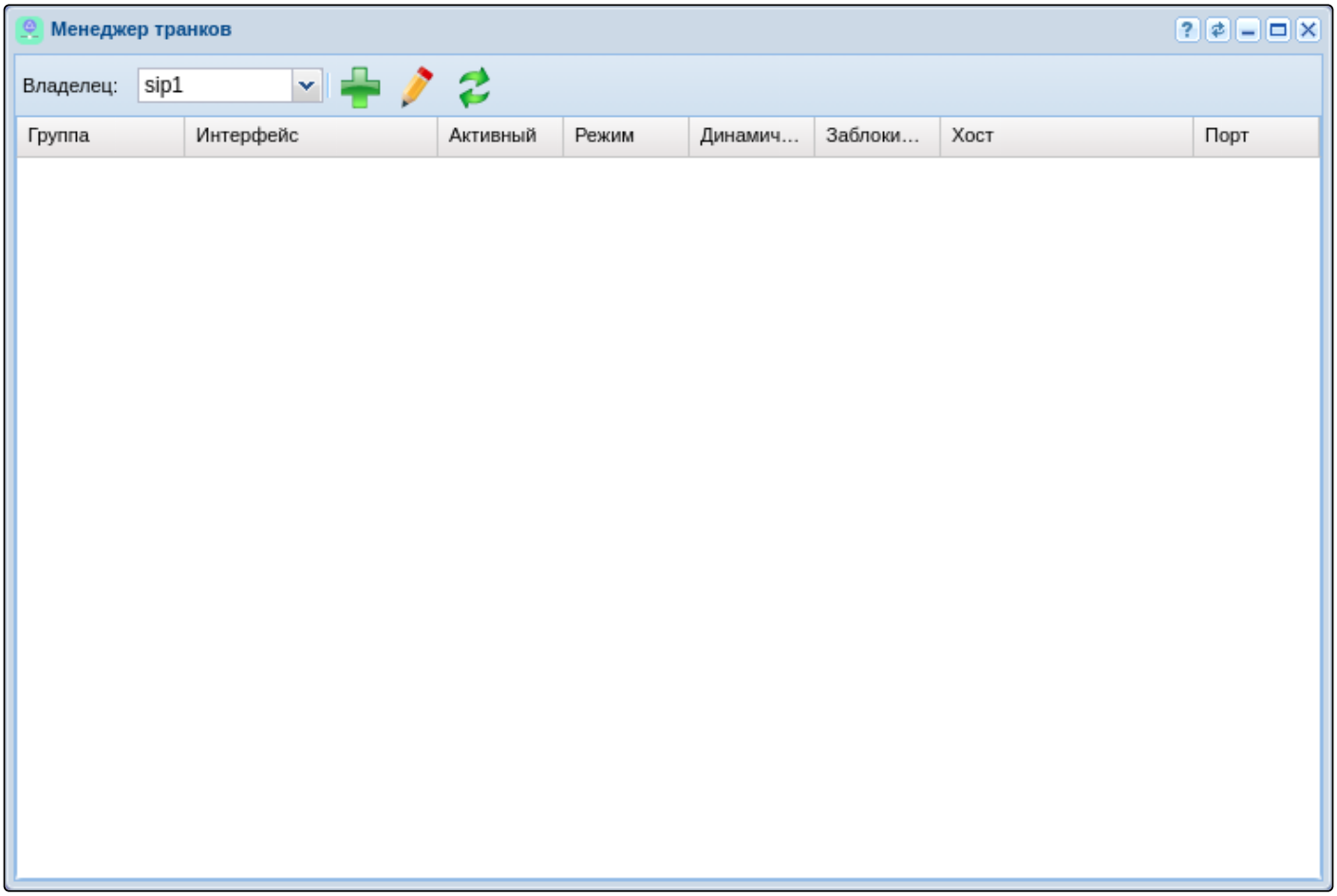


Рисунок 30 — Вид приложения "Менеджер транков"

Для определения транка в системе нужно нажать на кнопку *"Создать транк"*  , в появившемся диалоговом окне определить параметры:

- *Имя* — назначить название транка, по которому его можно идентифицировать в системе;
- *Контекст* — применить ранее созданный контекст маршрутизации test_name;
- *Группа* — выбрать группу интерфейсов, созданную при определении абонентов "test.group";
- *Группа IP-адресов (IP-set)* — созданная на домене группа адресов "test_set";
- *Регистрация* — если используется транковая, то нужно включить данный параметр, в примере она не используется;
- *Хост [:порт]* — IP-адрес назначения транка — 10.0.3.100;
- *Порт для приёма* — транспортный порт, на котором будет слушаться трафик с транка, соответствует порту, назначенному на группу IP-адресов.

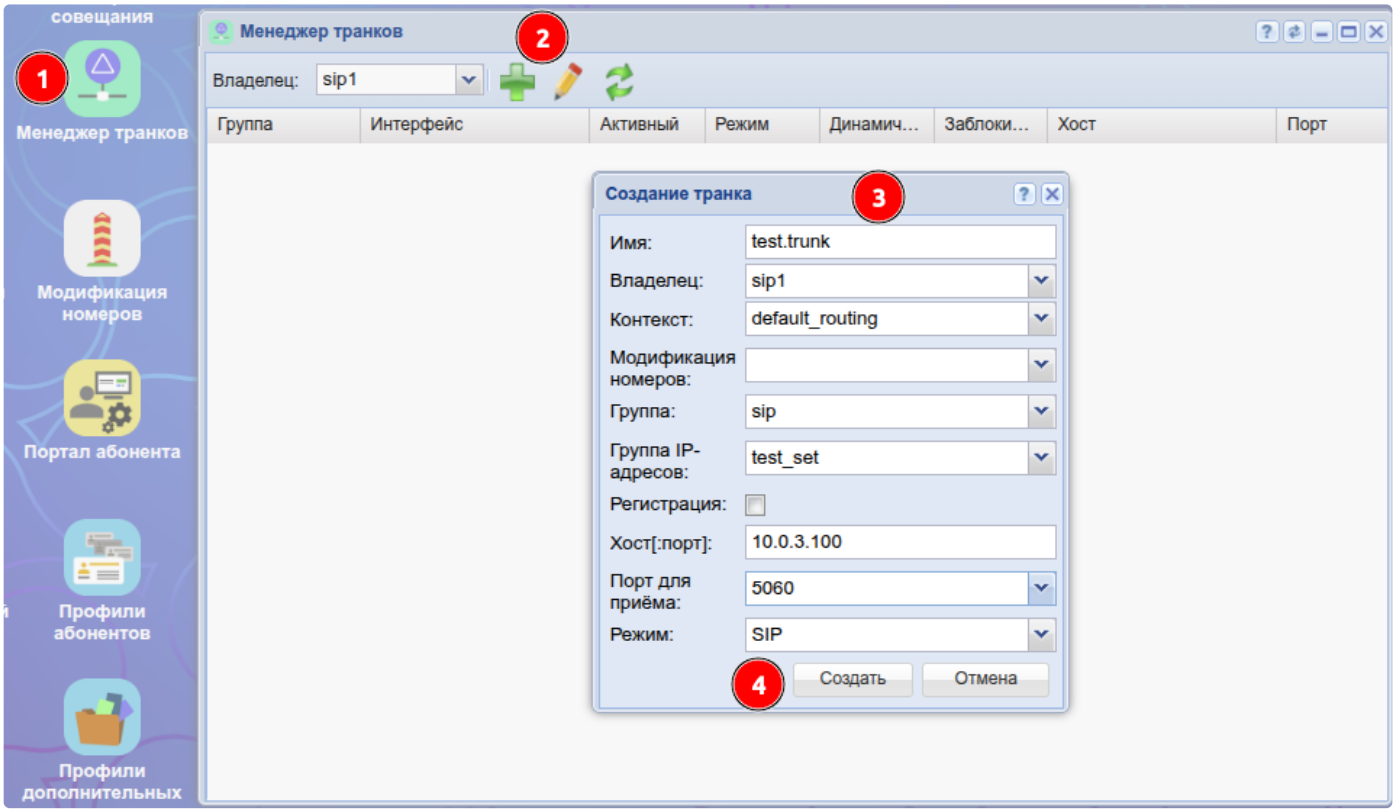


Рисунок 31 — Создание транка

12.6.10 Создание IVR-скрипта

Для создания IVR используется приложение "IVR-редактор" ("IVR editor").

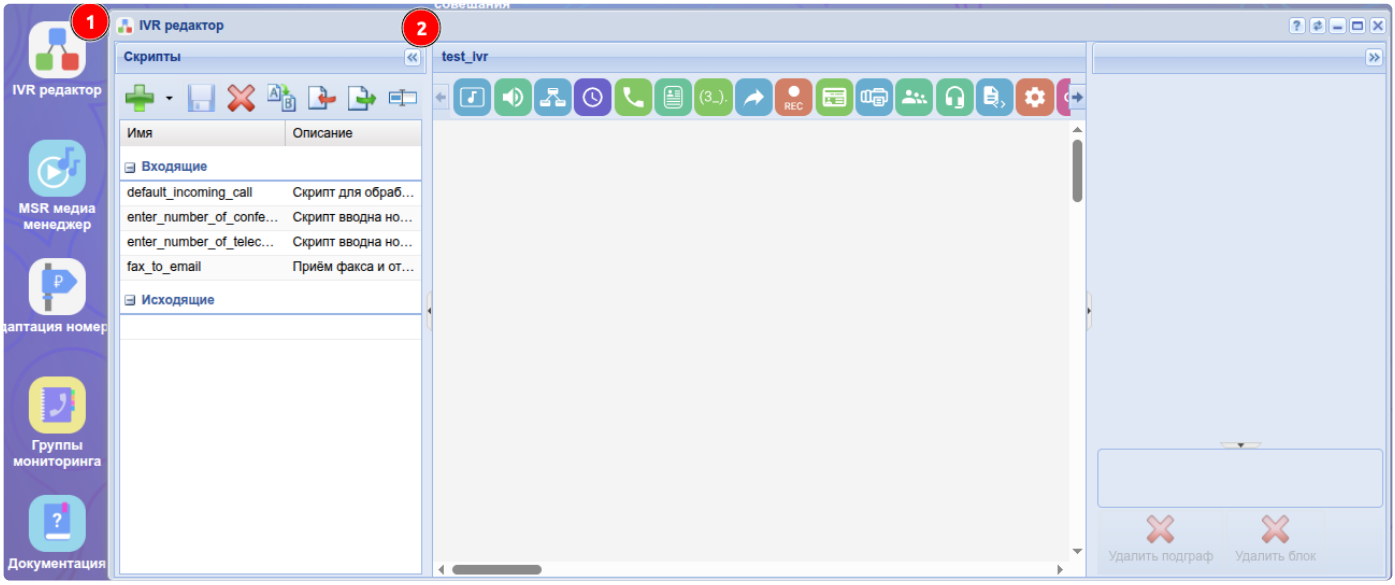


Рисунок 32 — Вид приложения "IVR-редактор"

Для создания скрипта нужно нажать на кнопку *"Добавить"* , выбрать тип скрипта (в данном случае будет скрипт на входящие вызовы), в диалоговом окне указать название скрипта, например "test_ivr".

После создания скрипта в основном окне редактора появится блок-схема.

В качестве примера составим скрипт, который при срабатывании проигрывает вызывающему абоненту заранее записанную фразу, а затем продолжает вызов.

Пример скрипта приведен на рисунках ниже. Здесь используются блоки Info и Dial.

- Блок Info — блок, который проигрывает сообщения до момента ответа пользователя. В качестве записанной фразы используется генератор тонов:

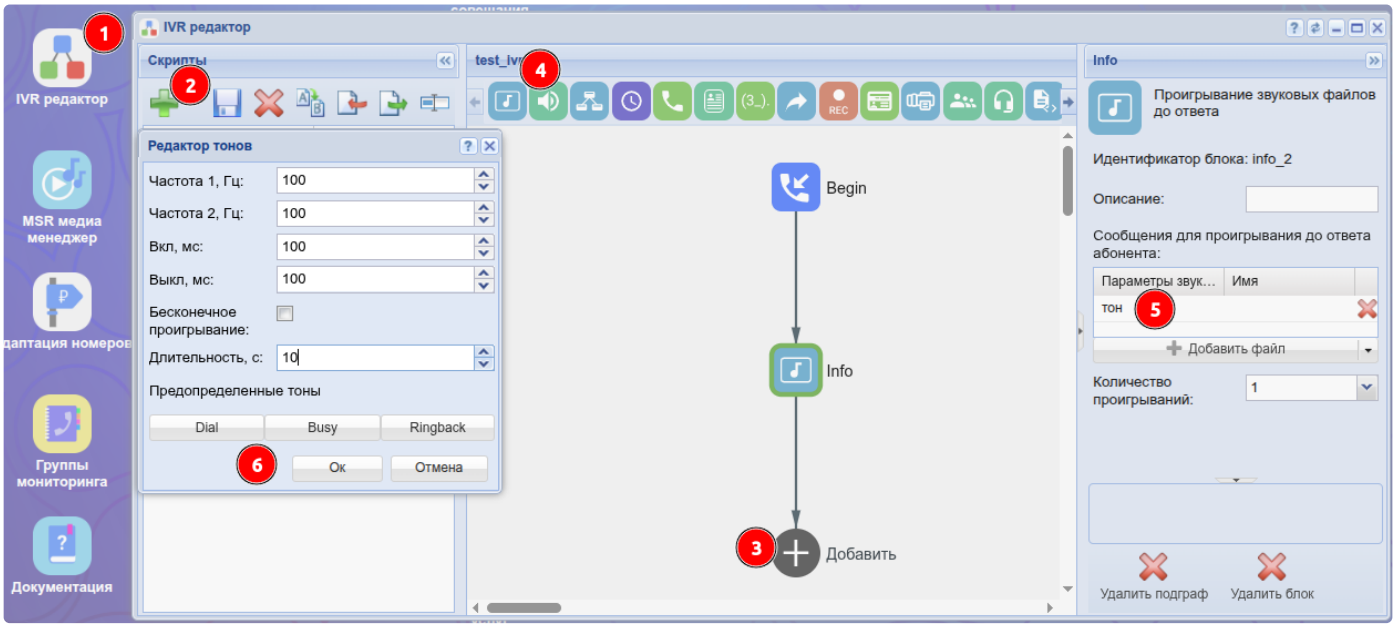


Рисунок 33 — Вид рабочей области IVR с настройками блока "Info"

- Блок dial — блок, который совершает вызов на заданный номер. Чтобы продолжить вызов, нужно применить предопределенную переменную CDPN:

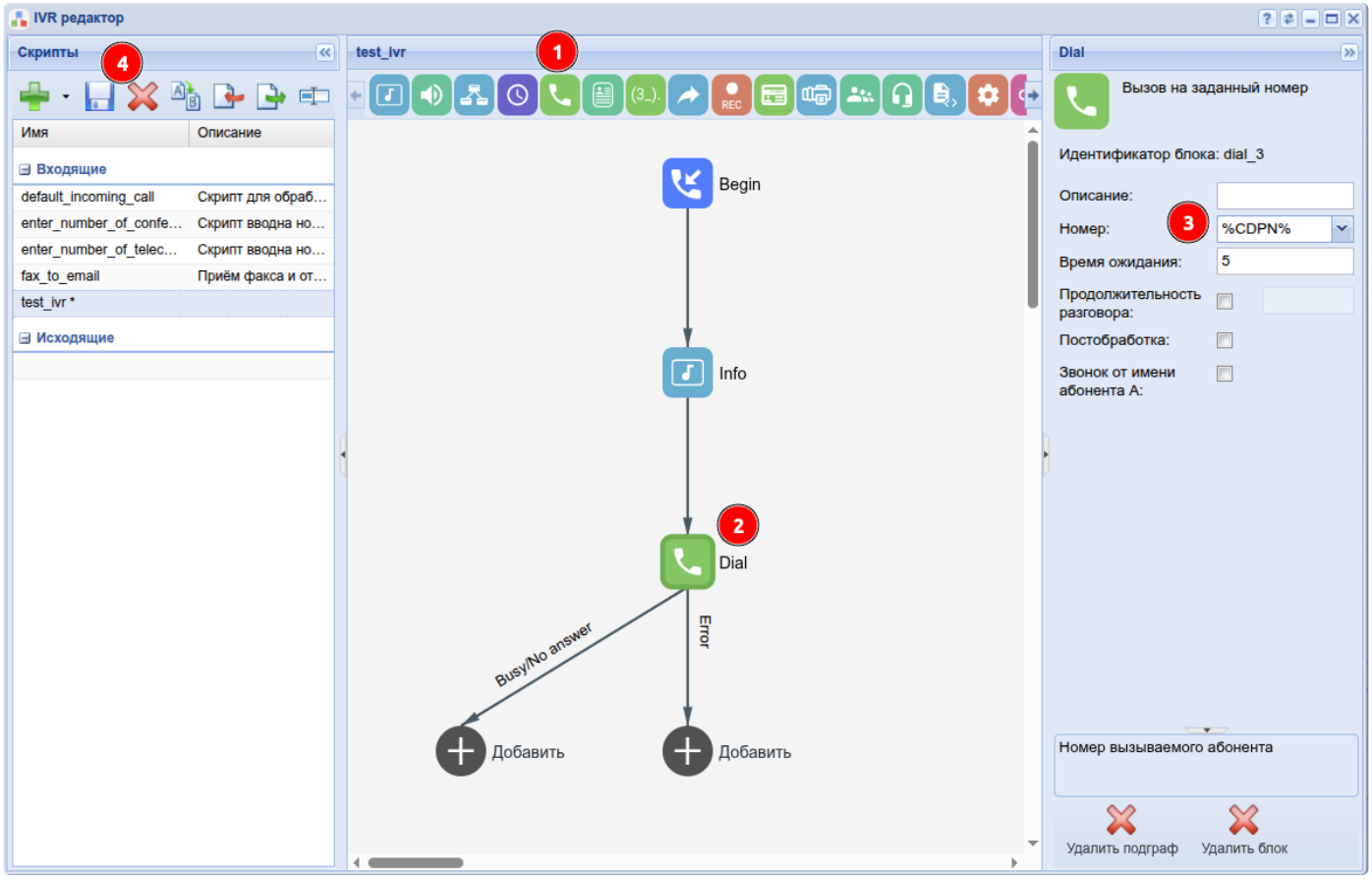


Рисунок 34 — Вид рабочей области IVR

12.6.11 Завершение настройки маршрутизации

Для завершения настройки маршрутизации нужно войти в приложение "Менеджер маршрутизации" ("Routing manager") и в правилах rule1 и rule3 настроить соответствующие результаты маршрутизации, для этого:

- rule1: в разделе "Результат" выберите подпункт "внешний". В таблице "Направления" в поле "Значение" добавьте созданный ранее транк, после чего сохраните правило .
- rule3: В разделе "Результат" выберите подпункт "ivr" и в поле скрипт добавьте созданный ранее ivr-скрипт, после чего сохраните правило и контекст .

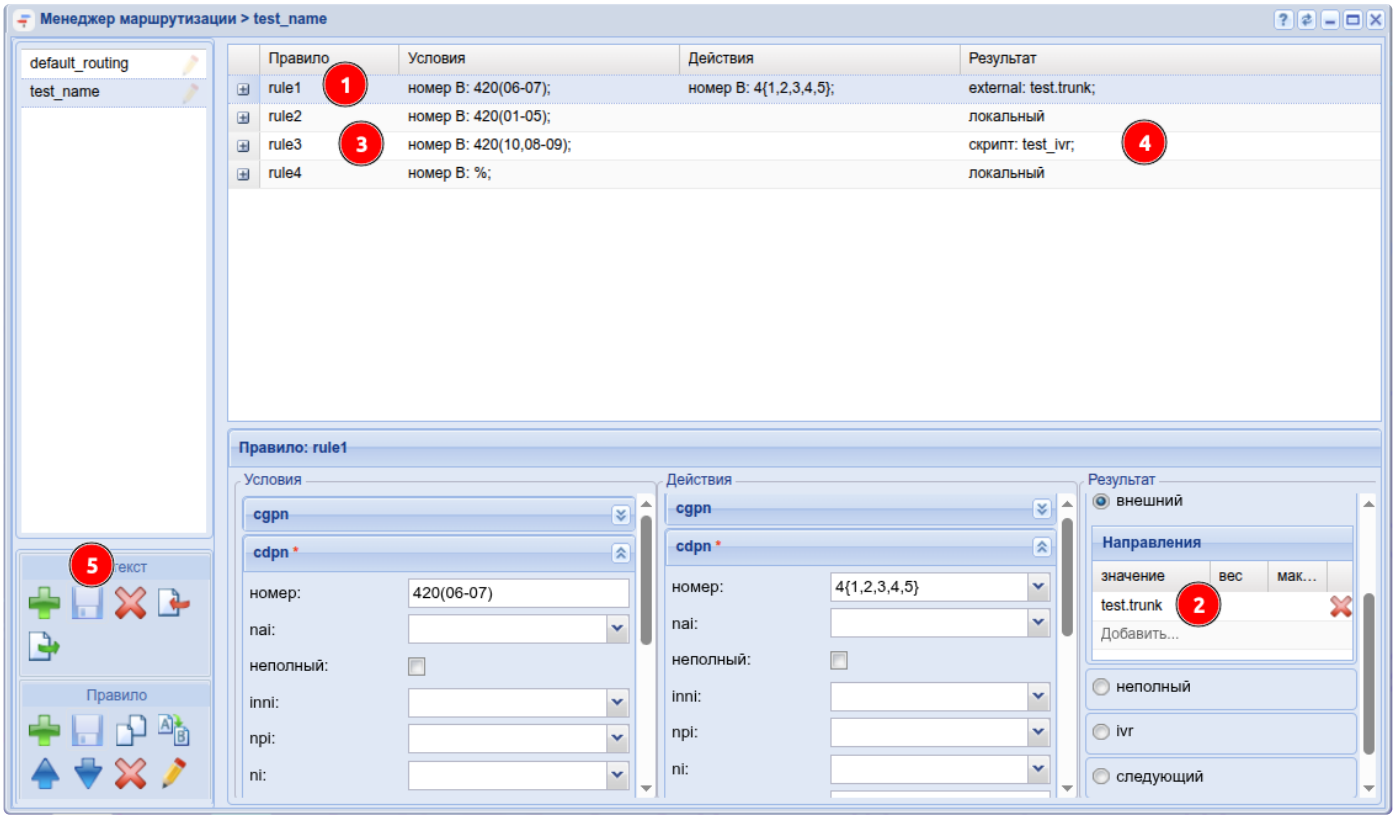


Рисунок 35 — Вид контекста маршрутизации

12.7 Первоначальная конфигурация системы через интерфейс командной строки (CLI)

пример первоначальной конфигурации через CoCon (CLI)

12.7.1 Добавление медиа-ресурсов в интерфейсе командной строки (CLI)

⚠ Для возможности использования медиаресурса и установления соединений , необходимо выполнить активизацию в команде создания медиаресурса - **/system/media/resource/declare** (параметр **"is active"=true**) или **/system/media/resource/set** (параметр **"active"=true**), если при создании активизация не была выполнена. Так же можно выполнить активизацию через веб интерфейс , приложение - сетевые окончания MSR (MSR-registrars).

Зайдите в CLI - CoCon:

```
ssh admin@localhost -p 8023
password: password
```

Медиаресурс — это описание параметров медиасервера, необходимых для работы с ним.

Медиаресурс — это описание параметров медиасервера, необходимых для работы с ним.

declare
Команда для объявления подключения медиасервера к ECSS-10.

Для работы в режиме предварительно настроенной системы можно заранее прописать медиаресурсы, информация о которых известна.

Для этого необходимо создать описание известных медиасерверов с указанием всех параметров, что позволить начать работу с медиасервером сразу после его регистрации.

Команда для создания конфигурации медиасервера:

Путь команды:

/system/media/resource/declare

Синтаксис:

declare <node> by_contact <contact> <zone> <site> <is active>

или

declare <node> by_iface <msr> <acc_id> <iface> <zone> <site> <is active>

Параметры:

<node> — имя ноды core;
<contact> — contact uri декларируемого медиаресурса;
<zone> — логическая зона обслуживания MSR;
<site> — идентификатор сети обслуживания MSR;
<is active> — состояние активности медиаресурса, значения true/false;
<msr> — имя MSR;
<acc_id> — номер идентификатор ресурса;
<iface> — название интерфейса.

Установить связь MSR- и Core-подсистем. Для этого выполнить команды:

```
/system/media/resource/declare core1@ecss1 contact net.20@msr.ecss1  default local true
```

```
/system/media/resource/declare core1@ecss2 contact net.20@msr.ecss2  default local true
```

```
/system/media/resource/declare core1@ecss1 contact net.20@msr.ecss2  default local true
```

```
/system/media/resource/declare core1@ecss2 contact net.20@msr.ecss1  default local true
```

результат

```
/system/media/resource/info core1@ecss1 net.20@msr.ecss1
  Media resource specific for node core1@ecss1:
┌─┬──────────────────┬──────────────────────────┐
│F│    Parameter     │          Value           │
├─┼──────────────────┼──────────────────────────┤
│r│ Contact          │ net.20@msr.ecss1         │
│r│ Status           │ registered               │
│r│ Cc-status        │ connected                │
│r│ Cc-status uptime │ 00:39:35                 │
│r│ Cc-id            │ db21af                   │
│r│ Address          │ 10.0.20.51:5700          │
│r│ Register time    │ 2025/12/02 16:43:07      │
│r│ Expired          │ 84                       │
│r│ Iface            │ net.20(10.0.20.51)       │
│ │ Active           │ true                     │
│ │ Zone             │ default                  │
│ │ Site             │ local                    │
│r│ MSR version      │ 3.18.1.0.10              │
│r│ MSR load         │ 0                        │
│r│ MSR perf coef    │ 1.0                      │
└─┴──────────────────┴──────────────────────────┘

Legend:
Column F - means Flag, values r - flag for readonly parameter
```

результат / список

```
/system/media/resource/list all
  All media resource selected list specific:
┌──────────────┬────────────┬──────────────┬──────────┬──────┬────────┬──────────┬──────────
──┬──────────┬──────────┐
│     Node     │    MSR     │     MSR      │    MSR   │ MSR  │ Cc-id  │ Cc-status │ Cc-uptime
│   Cc-address │ Iface  │   Iface    │ Active │  Zone   │ Site  │     Contact      │
   Status   │ Expired │
│              │            │   version    │ perf coef │ load │        │           │
│                  │  name  │    addr    │         │          │       │                  │
│          │
├──────────────┼────────────┼──────────────┼──────────┼──────┼────────┼──────────┼──────────
──┼──────────┼──────────┤
│ core1@ecss1 │ msr.ecss1 │ 3.18.1.0.10 │       1.0 │    0 │ db21af │ connected │ 00:40:22
│ 10.0.20.51:5700 │ net.20 │ 10.0.20.51 │ true   │ default │ local │ net.20@msr.ecss1 │
 registered │     156 │
│              │ msr.ecss2 │ 3.18.1.0.10 │       1.0 │    0 │ 263898 │ connected │ 00:40:22
│ 10.0.20.52:5700 │ net.20 │ 10.0.20.52 │ true   │ default │ local │ net.20@msr.ecss2 │
 registered │     156 │
```

```
| core1@ecss2 | msr.ecss1 | 3.18.1.0.10 |       1.0 |    0 | db21af | connected | 00:41:11
| 10.0.20.51:5700 | net.20 | 10.0.20.51 | true   | default | local | net.20@msr.ecss1 |
 registered |     106 |
|             | msr.ecss2 | 3.18.1.0.10 |       1.0 |    0 | 263898 | connected | 00:44:00
| 10.0.20.52:5700 | net.20 | 10.0.20.52 | true   | default | local | net.20@msr.ecss2 |
 registered |     177 |
└─────────────┴──────────┴─────────────┴────────────┴──────┴────────┴───────────┴──────────
──┴──────────────────┴─────────┴─────────────┴─────────┴──────────┴──────┴──────────────────┴──
──────┴──────────┘
```

12.7.2 Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI)

/system/media/registrar/ - команды управления регистратором для MSR медиа менеджера

/system/media/registrar/ - команды управления регистратором для MSR медиа менеджера

set

Команда для изменения общих и индивидуальных параметров.

Путь команды:

/system/media/registrar/set

Синтаксис настройки общих параметров:

set common <FIELD> <VALUE>

Параметры:

<FIELD> - имя настраиваемого параметра;

<VALUE> - значение настраиваемого параметра

Где

- common - тип выставляемого параметра;
- <common parameter> - общий параметр медиарегистратора для всех нод: registration-timeout;
- private - тип выставляемого параметра;
- <node name> - имя ноды на которой находится медиарегистратор;
- <private parameter> - параметр медиарегистратора, который выставляется на <node name>: is-active, listen-interface & listen-port.

Список параметров доступных для настройки:

- is-active - позволяет включить или выключить регистратор (false - деактивирует регистратор, регистратор перестает принимать SIP трафик);
- listen-interface - имя интерфейса, который использует регистратор;
- listen-port - порт, на котором регистратор принимает запросы на регистрацию (по умолчанию 5000);
- registration-timeout - таймер перерегистрации, диапазон значений от 60 до 86400.

В общих параметрах доступна настройка времени регистрации, по умолчанию установлено 60 секунд.

Пример:
Установить время регистрации 60 секунд:

```
/system/media/registrar/set common registration-timeout 60
Common parameter registration-timeout set to 60.
```

Синтаксис настройки индивидуальных параметров:

system/media/registrar/set private <CORE> <FIELD> <VALUE>

Параметры:

<CORE> - нода ядра, для которой настраивается Registrar-сервер;

<FIELD> - имя настраиваемого параметра;

<VALUE> - значение настраиваемого параметра.

Настраиваемые параметры:

- is-active - активен ли Registrar-сервер, принимает значения true | false;
- listen-ip - ip-адрес, на котором Registrar-сервер будет принимать соединения (по умолчанию 0.0.0.0);
- listen-port - сетевой порт, на котором Registrar-сервер будет принимать соединения (по умолчанию 5000).

Установить ipадрес для приема соединений от медиасервера:

выполняется в CoCon

```
/system/media/registrar/set private core1@ecss1 listen-ip 10.0.20.41
```

```
/system/media/registrar/set private core1@ecss1 listen-ip 10.0.20.41
Private parameter listen-ip set to 10.0.20.41 at node core1@ecss1.
```

выполняется в CoCon

```
/system/media/registrar/set private core1@ecss2 listen-ip 10.0.20.42
```

```
/system/media/registrar/set private core1@ecss2 listen-ip 10.0.20.42
Private parameter listen-ip set to 10.0.20.42 at node core1@ecss2.
```

Результат:

```
/system/media/registrar/info
 Common parameters:
┌──────────────────────────────┬──────┐
```

```
|         Parameter          | Value |
|----------------------------|-------|
| registration-timeout, sec  |   180 |
```

Private parameters:

```
|    Node      | Status  | Is-active | Listen-ip   | Listen-port |
|--------------|---------|-----------|-------------|-------------|
| core1@ecss1  | active  | true      | 10.0.20.41  |        5000 |
| core1@ecss2  | active  | true      | 10.0.20.42  |        5000 |
```

После того как все сервисы будут запущены, ноды какое-то время будут устанавливать связь. Как только все ноды загрузятся , при выполнении запроса **system-status**, последует вывод следующей информации:

```
/system-status
Checking...
|  |     Node       |          Release          |         Erlang nodes          |    Mnesia
nodes     |Uptime |
|--|----------------|---------------------------|-------------------------------|-----------
---------|-------|
|  |core1@ecss1     |ecss-core-3.18.1.0.1656    |core1@ecss1,core1@ecss2        |not running
|43m 43s|
|  |core1@ecss2     |ecss-core-3.18.1.0.1656    |core1@ecss1,core1@ecss2        |not running
|47m 5s |
|  |ds1@ecss1       |ecss-ds-3.18.1.0.1656      |ds1@ecss1,ds1@ecss2
|ds1@ecss1,ds1@ecss2  |43m 43s|
|  |ds1@ecss2       |ecss-ds-3.18.1.0.1656      |ds1@ecss1,ds1@ecss2
|ds1@ecss1,ds1@ecss2  |47m 4s |
|  |md1@ecss1       |ecss-mediator-3.18.1.0.1656|md1@ecss1,md1@ecss2
|md1@ecss1,md1@ecss2  |43m 43s|
|  |md1@ecss2       |ecss-mediator-3.18.1.0.1656|md1@ecss1,md1@ecss2
|md1@ecss1,md1@ecss2  |47m 5s |
|  |mycelium1@ecss1|ecss-mycelium-3.18.1.0.1656|mycelium1@ecss1,mycelium1@ecss2|not running
|43m 43s|
|  |mycelium1@ecss2|ecss-mycelium-3.18.1.0.1656|mycelium1@ecss1,mycelium1@ecss2|not running
|47m 4s |
|  |sip1@ecss1      |ecss-pa-sip-3.18.1.0.1656  |sip1@ecss1,sip1@ecss2
|sip1@ecss1,sip1@ecss2|43m 43s|
|  |sip1@ecss2      |ecss-pa-sip-3.18.1.0.1656  |sip1@ecss1,sip1@ecss2
|sip1@ecss1,sip1@ecss2|47m 5s |
|  |                |                           |                               |
---------|-------|

All services are started.

  Active media resource selected list specific:
|    Node      |    MSR     |     MSR      | Cc-status | Cc-uptime |
|              |            |   version    |           |           |
|--------------|------------|--------------|-----------|-----------|
| core1@ecss1 | msr.ecss1 | 3.18.1.0.10 | connected | 00:42:49  |
```

```
|              | msr.ecss2 | 3.18.1.0.10 | connected | 00:42:48  |
| core1@ecss2  | msr.ecss1 | 3.18.1.0.10 | connected | 00:43:38  |
|              | msr.ecss2 | 3.18.1.0.10 | connected | 00:46:27  |
└──────────────┴───────────┴─────────────┴───────────┴───────────┘

Alarms list on md1:
┌──────────────┬──────────────┬──────────────┬──────────────────────┬──────────────────────┐
└──────────────┴──────────────┴──────────────┴──────────────────────┴──────────────────────┘
|     Date     |   Severity   |   Location   |       Cause          |       Class          |
     Instance      |                          Message                          |
├──────────────┼──────────────┼──────────────┼──────────────────────┼──────────────────────┤
└──────────────┴──────────────┴──────────────┴──────────────────────┴──────────────────────┘

Selected: 0
Total: 0
```

В примере выше видно, что ноды вошли в кластер и *MSR* зарегистрировались на ноде *ecss-core*.

Создание и настройка группы SIP-IP-адресов (IP-set)

IP-set — совокупность одного или нескольких IP-адресов из одной подсети и списка слушающих UDP/TCP-портов, которые будут открыты SIP-адаптером на данных адресах.

Произведите настройку адаптера SIP согласно ТЗ (команды выполнить в CoCon):

выполняется в CoCon

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss1 ip = 10.0.20.31
```

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node =
sip1@ecss1 ip = 10.0.20.31
Property "ip_set" successfully changed from:

to
test_set: no ports set
test_set: sip1@ecss1 10.0.20.31
test_set: dscp 0.
```

выполняется в CoCon

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss2 ip = 10.0.20.32
```

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node =
sip1@ecss2 ip = 10.0.20.32
Property "ip_set" successfully changed from:
test_set: no ports set
test_set: sip1@ecss1 10.0.20.31
test_set: dscp 0
to
test_set: no ports set
test_set: sip1@ecss1 10.0.20.31
test_set: sip1@ecss2 10.0.20.32
test_set: dscp 0.
```

выполняется в CoCon

```
/cluster/adapter/sip1/sip/network/set ip_set test_set listen-ports list = [5060]
```

```
/cluster/adapter/sip1/sip/network/set ip_set test_set listen-ports list = [5060]
Property "ip_set" successfully changed from:
```

```
test_set: no ports set
test_set: sip1@ecss1 10.0.20.31
test_set: sip1@ecss2 10.0.20.32
test_set: dscp 0
   to
test_set: 5060
test_set: sip1@ecss1 10.0.20.31
test_set: sip1@ecss2 10.0.20.32
test_set: dscp 0
```

12.7.3 Создание Домена и подключение IP-set

Создайте Домен (виртуальную АТС) и присвойте ему созданную выше группу (IP-set) настроек SIP-адаптера (команды выполняются в CoCon):

выполняется в CoCon

```
/domain/declare test_domain --add-domain-admin-privileges --add-domain-user-privileges
```

```
/domain/declare test_domain --add-domain-admin-privileges --add-domain-user-
privileges
New domain test_domain is declared
```

выполняется в CoCon

```
/domain/test_domain/sip/network/set ip_set [test_set]
```

```
/domain/test_domain/sip/network/set ip_set [test_set]

Property "ip_set" successfully changed from:

[]
    to
["test_set"].
```

12.7.4 Добавление услуг

Разрешить оператору Домена (виртуальной АТС) "test_domain" управлять всеми услугами :

выполняется в CoCon

```
/cluster/storage/ds1/ss/access-list add test_domain *
```

```
/cluster/storage/ds1/ss/access-list add test_domain *
Supplementary services successfully added in the domain test_domain.
```

Для возможности применения пакета услуг у абонентов сразу после создания, необходимо настроить политику применения лицензионных пакетов услуг (детальную информацию вы можете посмотреть здесь)

Создать политику применения лицензии с именем -"new_sub"

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/declare new_sub "Задаются при создании абонента"
```

Задать имя пакета услуг дополнительных сервисов, для этого использовать следующие команды :

выполняется в CoCon user/pw admin/password

```
/cluster/storage/ds1/licence/show-licence 1
```

нас интересует следующая информация :

```
/cluster/storage/ds1/licence/show-licence 1
SSW ID                                                  < Имя SSW ID >
Description                                             Лицензия для тестирования.

. . .

Supplementary Services licence package(s)
 name                                                   'ECSS-FULL+'
 limit                                                  600000
 description                                            "Весь пакет услуг"
 SS list                                                [1,2,3,4,5]

. . .
```

имя пакета услуг в примере "ECSS-FULL+".
Добавить в созданную политику пакет лицензионных услуг с именем "ECSS-FULL+", именно так он описан в лицензии:

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/package-add new_sub ECSS-FULL+
```

Пакет добавлен, но в настоящий момент политика не активна (не может быть применена к абоненту). Для активации политики new_sub выполнить следующую команду:

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/activate new_sub
```

Теперь при создании абонентов можем использовать дополнительные услуги.

13 Установка ECSS10 3.18 на Astra Linux, в закрытом контуре, на одиночный сервер

13.1 Первоначальная установка системы на одном сервере

13.1.1 Исходные данные

Конфигурация системы с одним сервером

Исходные данные

В данном разделе использованы специфичные для данного продукта термины, ознакомиться с ними можно здесь.
Техническое задание:
Требуется интеграция программного коммутатора 5 класса ECSS-10 (SSW) на 1-м физическом сервере с поддержкой SIP со следующими параметрами на нагрузку:

- Максимальное количество абонентов 15000 (MUL — Max user limit);
- Максимальное количество одновременных соединений 2000 (MCL — Max call limit);
- Количество сетевых интерфейсов ethernet — 3/5.

По техническому заданию требуется определить аппаратную платформу.

Полоса пропускания СПД

Требуемая полоса пропускания СПД **не менее** 1000Mb/s

Пример составления аппаратных требований:

Пример составления аппаратных требований:

Пример разнесения составляющих по адресному пространству				
Имя сервера (хоста)	**Роль**	**Интерфейс**	**Адрес/Маска**	**Порт**
Статические адреса программного коммутатора				
ecss1	Адрес сервера	eth0	192.168.56.40/24	2000
ecss1	Адрес протокольного адаптера	int.20 (voip)	10.0.20.40/24	5000
ecss1	Адрес шлюза	eth0	192.168.56.1	-
Виртуальные адреса программного коммутатора				
ecss1	Адрес ядра (*ecss-core*)	lo	10.0.20.40/24	5000

ecss1	Адрес медиа-сервера (*ecss-media-server (MSR)*)	lo	10.0.20.40/24	5000
Заполнить данные параметры для SSW из проектного технического задания				
Имя сервера (хоста)	**Роль**	**Интерфейс**	**Адрес/Маска**	**Порт**
Статические адреса программного коммутатора				
ecss1	Адрес сервера			2000
ecss1	Адрес протокольного адаптера			5000
ecss1	Адрес шлюза			-
Виртуальные адреса программного коммутатора				
ecss1	Адрес ядра (*ecss-core*)	lo		5000
ecss1	Адрес медиа-сервера (*ecss-media-server (MSR)*)	lo		5000

Включение в сеть

Пример на рисунке ниже.

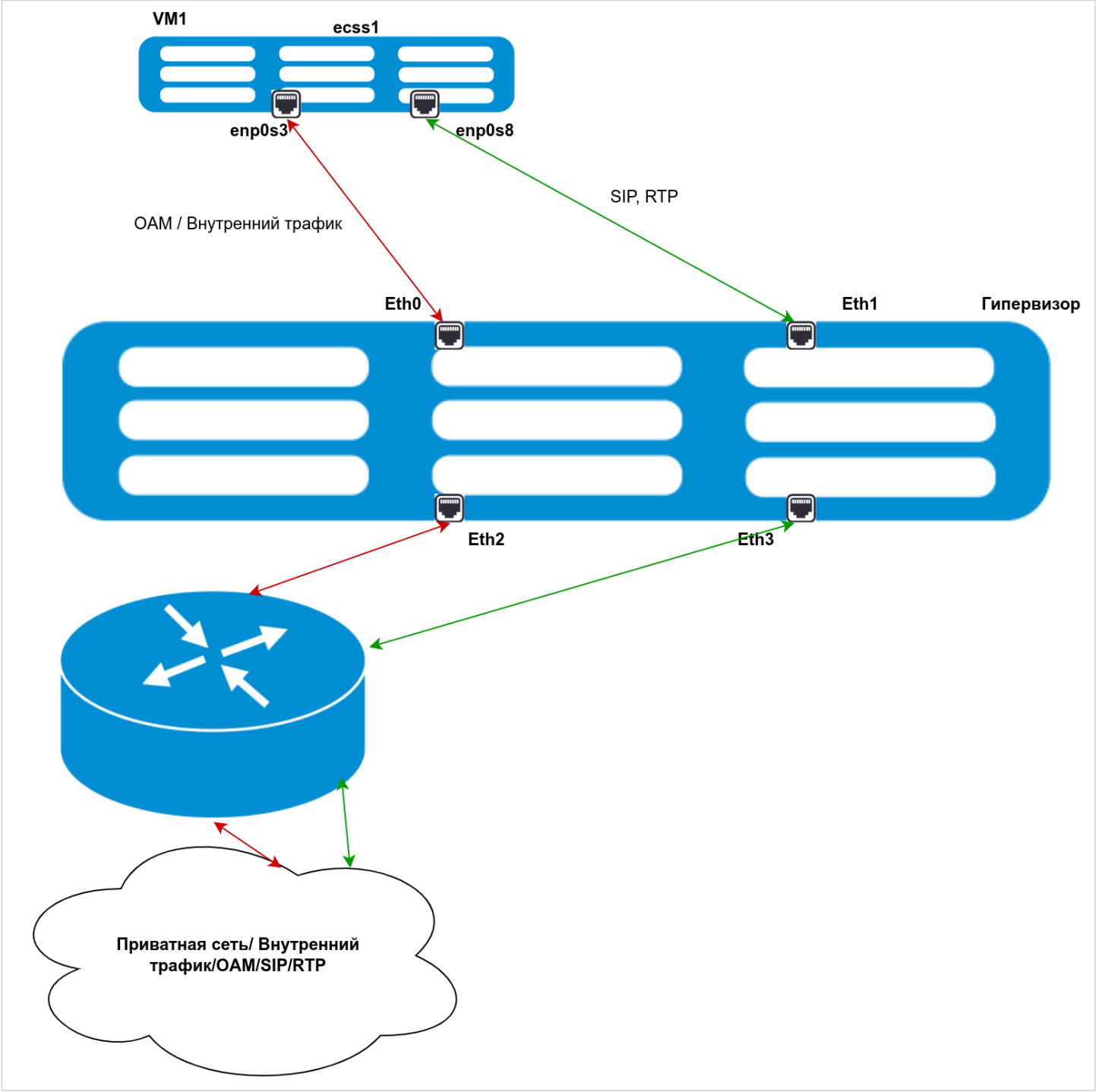


Рисунок 1 – Схема организации связи

13.1.2 Последовательность установки SSW ECSS10 без резервирования с одним сервером

Установка SSW ECSS10 состоит из 2-х основных частей :

- Подготовительной - Установка ОС Astra Linux 1.8, обновлений пакетов ОС Astra Linux 1.8 , установка дополнительных пакетов программ, подготовка сетевых интерфейсов, /etc/hosts, ssh-keygen
- Установка приложений Eltex ECSS10

Подготовительный этап

В этом разделе приведено описание инсталляции операционной системы, а также необходимых и дополнительных пакетов. Система ECSS-10 версии 3.18 работает под управлением ОС Astra Linux 1.8.

Предварительные требования

- Установочный загрузочный носитель с дистрибутивом операционной системы ;
- Подготовленный сервер с обновленным BIOS, ILO (если есть), подключенная сеть для доступа ;
- Выставленный первый приоритет загрузки с установочного носителя — USB Flash или CD/DVD в BIOS;
- Достаточный объем дискового пространства и памяти в соответствии с проектом.

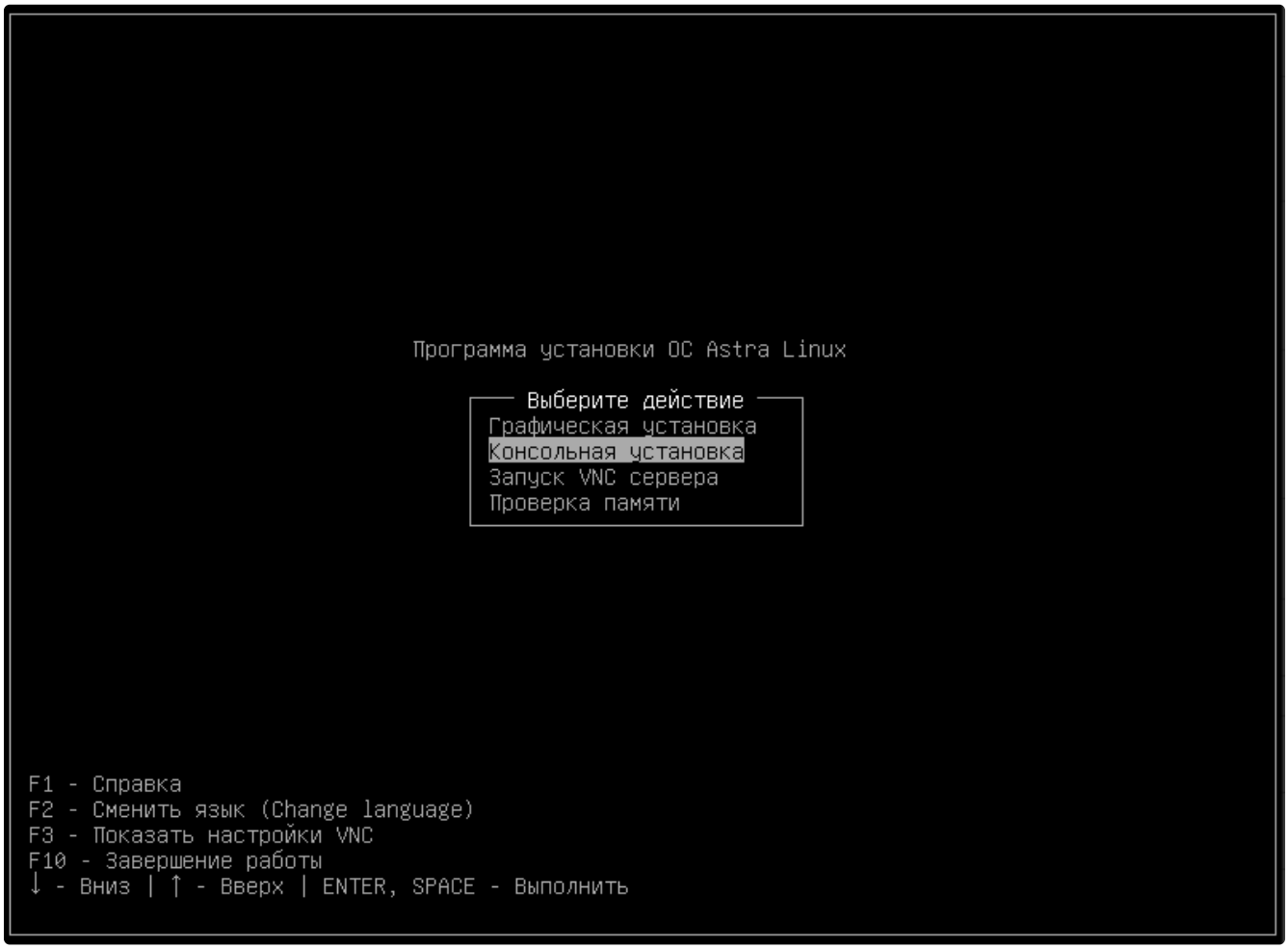
13.1.3 Установка ОС

Для установки ОС необходимо выполнить следующее:

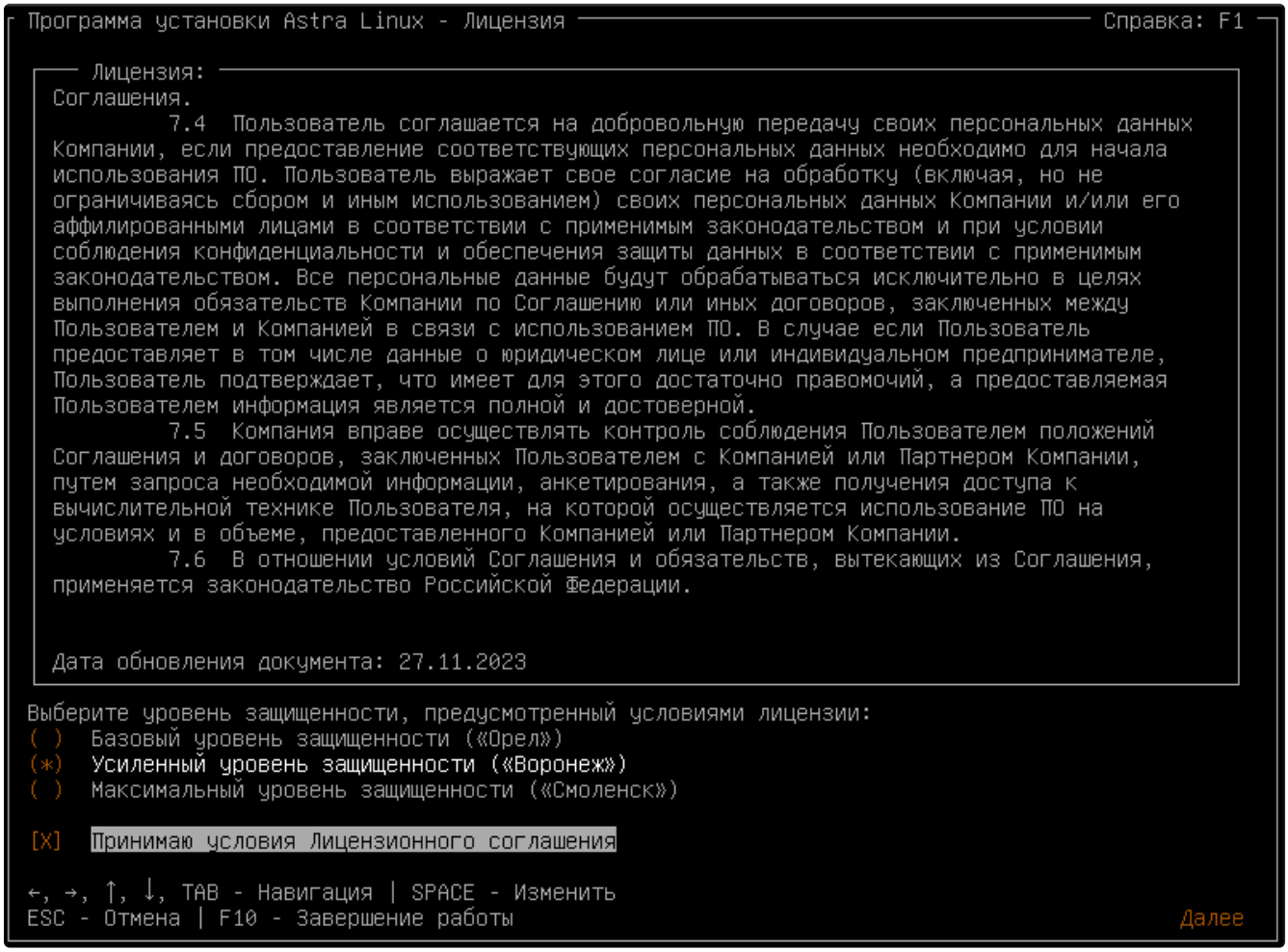
- После загрузки с установочного носителя выбрать "Графическая установка" или "Установка".
- Согласится с лицензионным соглашением.
- Выбрать раскладку клавиатуры.
- Выбрать сетевой интерфейс для подключения к сети Интернет.
- Установить имя компьютера (рекомендованы ecss1)
- Настроить пользователя- администратора.
- Создать партиции на диске в соответствии с таблицей 1 (разметка диска - Вручную).
- В процессе установке , **не создавать раздел swap** . При напоминании о пропуске раздела , **подтвердить** , что установка выполняется без создания swap. (см рис. 1)
- Установить часовой пояс.
- Установить OpenSSH server (активизировать пункт : "Средства удаленного подключения SSH" см рис. 2).
- Выбрать уровень защищенности, который используется в вашем проекте (в примере будем использовать вариант "Воронеж" см рис.3)
- Дополнительные настройки ОС - по умолчанию(см рис.4)
- Перезагрузить успешно установленную ОС

Установка ОС Astra Linux 1.8

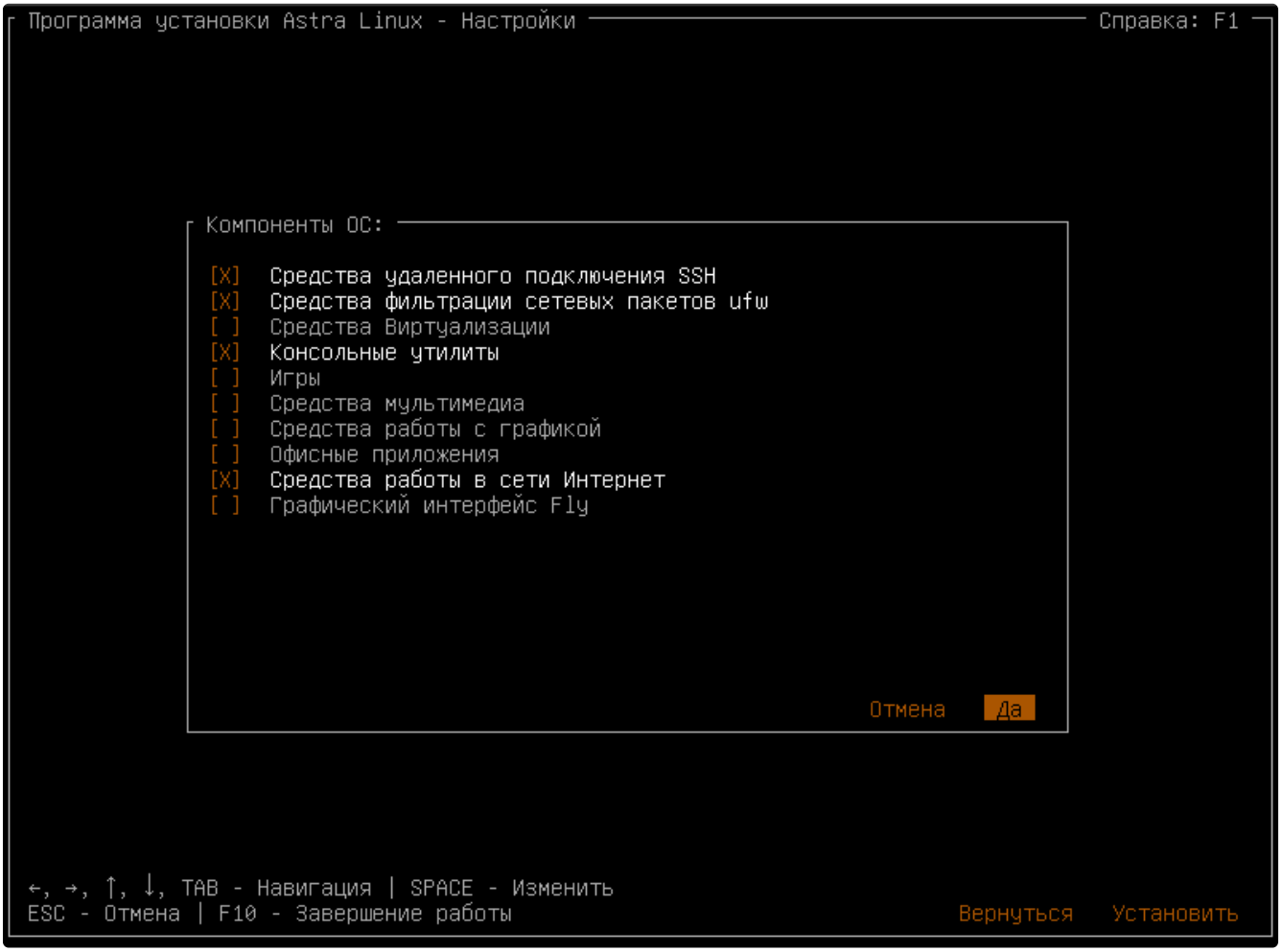
Выбрать режим установки



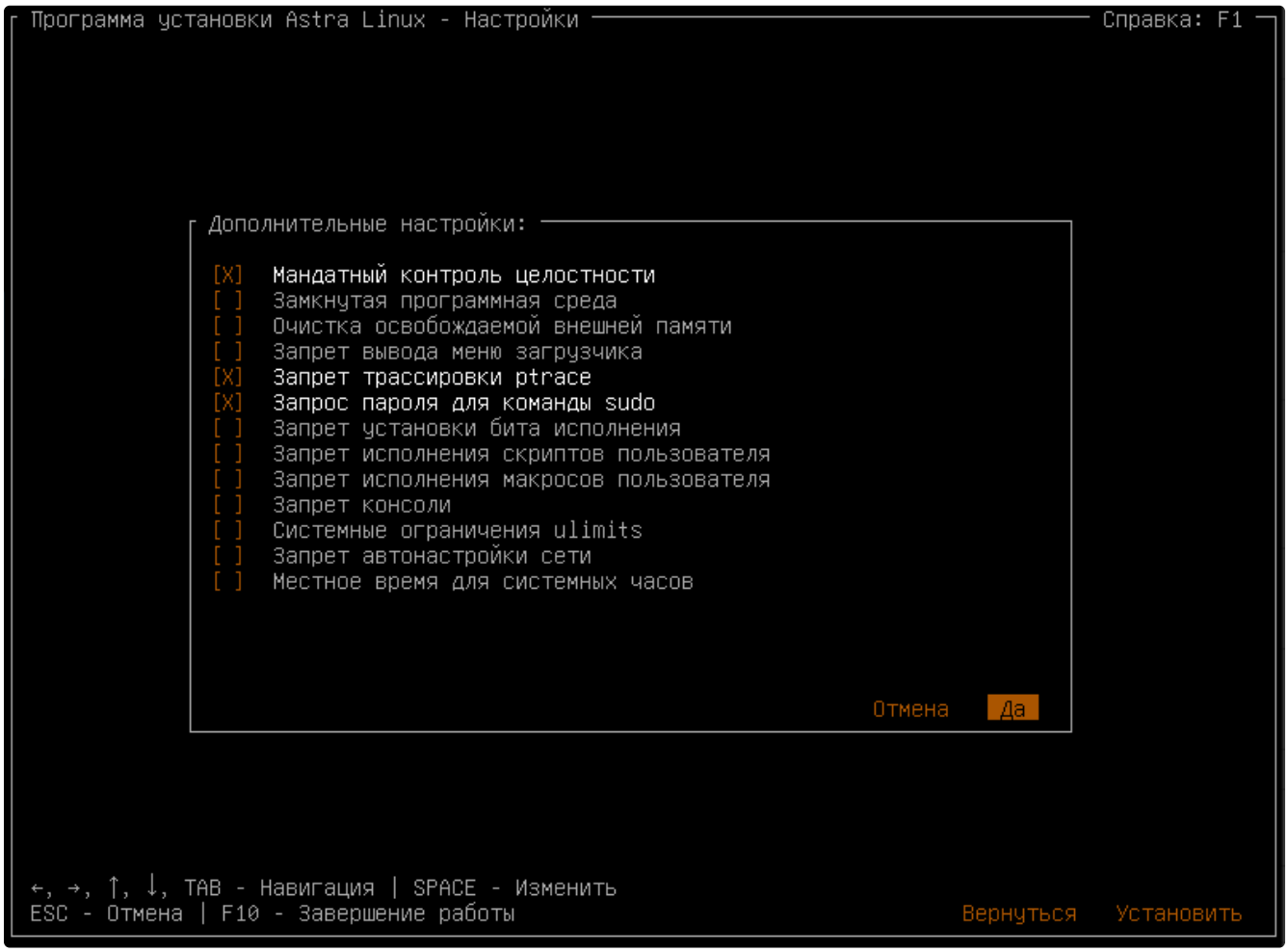
Согласится с Лицензионным Соглашением


Программа установки Astra Linux - Лицензия
Справка: F1
Лицензия:
Соглашения.
7.4 Пользователь соглашается на добровольную передачу своих персональных данных Компании, если предоставление соответствующих персональных данных необходимо для начала использования ПО. Пользователь выражает свое согласие на обработку (включая, но не ограничиваясь сбором и иным использованием) своих персональных данных Компании и/или его аффилированными лицами в соответствии с применимым законодательством и при условии соблюдения конфиденциальности и обеспечения защиты данных в соответствии с применимым законодательством. Все персональные данные будут обрабатываться исключительно в целях выполнения обязательств Компании по Соглашению или иных договоров, заключенных между Пользователем и Компанией в связи с использованием ПО. В случае если Пользователь предоставляет в том числе данные о юридическом лице или индивидуальном предпринимателе, Пользователь подтверждает, что имеет для этого достаточно правомочий, а предоставляемая Пользователем информация является полной и достоверной.
7.5 Компания вправе осуществлять контроль соблюдения Пользователем положений Соглашения и договоров, заключенных Пользователем с Компанией или Партнером Компании, путем запроса необходимой информации, анкетирования, а также получения доступа к вычислительной технике Пользователя, на которой осуществляется использование ПО на условиях и в объеме, предоставленного Компанией или Партнером Компании.
7.6 В отношении условий Соглашения и обязательств, вытекающих из Соглашения, применяется законодательство Российской Федерации.
Дата обновления документа: 27.11.2023
Выберите уровень защищенности, предусмотренный условиями лицензии:
( ) Базовый уровень защищенности («Орел»)
(*) Усиленный уровень защищенности («Воронеж»)
( ) Максимальный уровень защищенности («Смоленск»)
[X] Принимаю условия Лицензионного соглашения
←, →, ↑, ↓, TAB - Навигация | SPACE - Изменить
ESC - Отмена | F10 - Завершение работы
Далее

Настройки



Настойки (продолжение1)



Настройки (продолжение2)


Программа установки Astra Linux - Настройки
Справка: F1
Региональные настройки
Переключения языка: Ctrl+Shift
Системный язык: Русский
Часовой пояс: UTC+07:00 Россия (Новосибирск)
Дополнительные настройки: 27.11.2025, 09:12, NTP service d...
Настройки авторизации
Имя компьютера: astra-ecss
Имя администратора: eltex
Пароль: ***
Подтверждение пароля: ***
Настройки пароля GRUB: Совпадает с администраторским
Другие настройки
Компоненты ОС: Средства фильтрации сетевых паке...
Дополнительные настройки: Мандатный контроль целостности, ...
Дополнительные пакеты: Не выбраны
Ядро для установки: linux-6.1-generic (Рекомендуется)
Разметка диска: Использовать все пространство на...
[X] Перезагрузить компьютер после завершения установки
←, →, ↑, ↓, TAB - Навигация | SPACE - Изменить
ESC - Отмена | F10 - Завершение работы
Вернуться
Установить

Установка

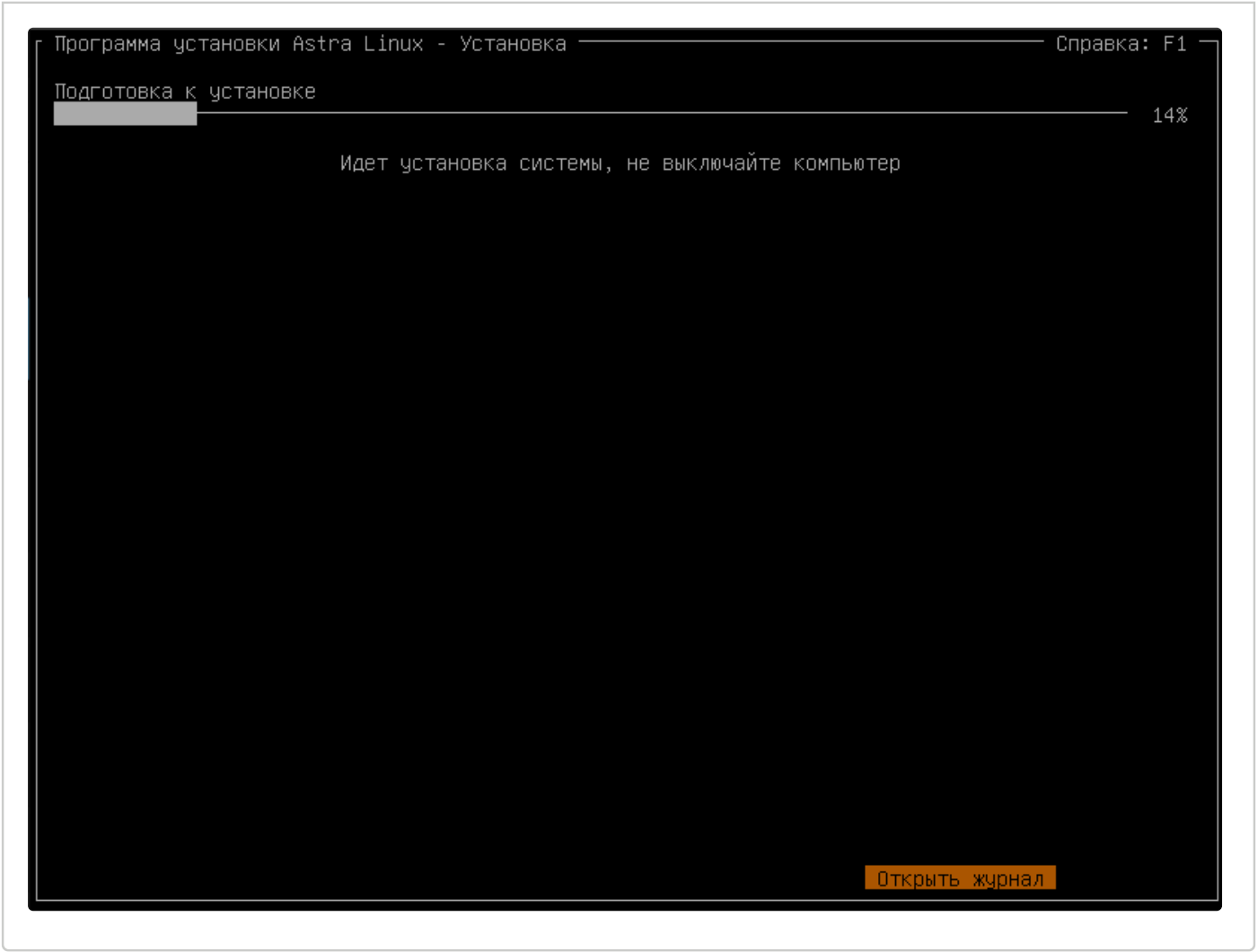


Таблица 1 — Вариант размещения информации в файловой системе на физических носителях для серверов

1	Загрузочный раздел операционной системы (создается автоматически)	boot	raid 1:hdd1,hdd2	boot	/boot	ext4	1 Gb	Первичный
2	Корневой раздел операционной системы	root	raid 1:hdd1,hdd2	root	/	ext4	30 Gb	Логический
3	Информация локальных баз данных	mnesia	raid 1:hdd1, hdd2	mnesia	/var/lib/ecss	ext4	30 Gb	Логический
4	Распределенная БД для хранения медиаресурсов	glusterfs	raid 1:hdd1, hdd2 или hdd3	glusterfs	/var/lib/ecss/restfs	ext4	Max Gb	Логический

5	Журналы функционирования подсистем ОС	log	raid 1:hdd1,hdd2 или hdd3	log	/var/log	ext4	20 Gb	Логический
6	Журналы функционирования подсистем ECSS	ecss_log	raid 1:hdd1,hdd2 или hdd3	ecss_log	/var/log/ecss	ext4	20 Gb	Логический
7	Базы данных	ecss_db	raid 1:hdd1,hdd2 или hdd3	ecss_db	/srv/ecss/ecss-postgres-bdr-ssw/	ext4	100–400 Gb*	Логический
8	Файлы пользователя	home	raid 1:hdd1,hdd2 или hdd3	home	/home	ext4	10 Gb	Логический

⚠ * Рекомендуемое значение для серий **Light, Light+, Midi** — 100 Gb. Рекомендуемое значение для серии **Heavy** — 200 Gb, **Super Heavy** — 400 Gb.

Для работы системы необходимо как **минимум** 256 Gb свободного пространства.

На серверах системы необходимо настроить параметр **"hostname".**

На всех серверах системы желательно указать одинаковое имя пользователя (любое, кроме **ssw**). Лицензия ECSS-10 привязывается к ключу eToken/ruToken и к имени компьютера (**hostname**). Системный пользователь **ssw** создается при инсталляции пакета **ecss-user**.

✔ **Рекомендуемое** значение hostname — **ecss1**;

13.1.4 Проверка установки ОС

Проверка в основном сводится к правильности создания разделов диска и наличия доступа по ssh.

Для вывода информации о состоянии дискового пространства введите команду **df -h**. Она покажет общее и занятое место в разделах. Размеры разделов должны соответствовать проекту и введенным значениям при установке.

Для проверки доступа по ssh с машины, находящейся в одной подсети со вновь установленным сервером, нужно выполнить команду:

```
ssh <user>@<IP_ecss>
```

где:

- <user> — имя пользователя, заданного при установке(для примера eltex);
- <IP_ecss> — IP-адрес хоста, заданного при установке.

13.1.5 Настройка /etc/hosts

посмотреть можно командой ***cat /etc/hosts***
изменить/добавить командой ***sudo nano /etc/hosts***

```
sudo nano /etc/hosts
```

ecss1:

```
127.0.0.1 localhost
10.0.10.40 ecss1
```

13.2 Конфигурация репозиториев

После установки для обновления будет использоваться только образ DVD диска, загрузка с репозитория отключена. Давайте разрешим обновление пакетов с локального репозитория Eltex.

Копируем образ репозитория Eltex в локальную директорию.

на удаленном хосте

```
scp eltex_repo_latest.tar.gz eltex@10.0.20.40:/home/eltex
```

на удаленном хосте

```
scp eltex_repo_gpg.key eltex@10.0.20.40:/home/eltex
```

И распаковываем архив.

пример:

```
tar -xzf eltex_repo_latest.tar.gz
```

перемещаем директорию с Eltex пакетами в /var/spool/:

```
sudo mv eltex_repo_latest /var/spool/
```

добавляем ключ :

```
sudo apt-key add eltex_repo_gpg.key
```

Редактируем source.list:

```
echo "deb file:/var/spool/eltex_repo 1.8_x86-64 contrib main non-free" | sudo tee -a /etc/apt/
sources.list
```

13.2.1 Изменение приоритета репозиториев

Для установки Erlang, Nodejs и Elixir из репозитория Eltex, необходимо изменить приоритет репозиториев для данных пакетов.

Добавить в файл /etc/apt/preferences.d/erlang строки:

```
sudo sh -c "echo -e 'Package: erlang*\nPin: version 2:24.3*\nPin-Priority: 1000' >> /etc/apt/
preferences.d/erlang"
```

Результат: cat /etc/apt/preferences.d/erlang

```
Package: erlang*
Pin: version 2:24.3*
Pin-Priority: 1000
```

Добавить в файл /etc/apt/preferences.d/nodejs строки:

```
sudo sh -c "echo -e 'Package: nodejs*\nPin: version 14.20.0-1nodesource1*\nPin-Priority: 1000'
>> /etc/apt/preferences.d/nodejs"
```

Результат: cat /etc/apt/preferences.d/nodejs

```
Package: nodejs*
Pin: version 14.20.0-1nodesource1*
Pin-Priority: 1000
```

Добавить в файл /etc/apt/preferences.d/elixir строки:

```
sudo sh -c "echo -e 'Package: elixir*\nPin: version 1.14*\nPin-Priority: 1000' >> /etc/apt/
preferences.d/elixir"
```

Результат: cat /etc/apt/preferences.d/elixir

```
Package: elixir*
Pin: version 1.14*
Pin-Priority: 1000
```

Выполнить обновление:

```
sudo apt update
```

13.2.2 "Оптимизация" Операционной Системы

Выставить параметры ОС в режим производительности

Используем утилиту cpufrequtils.

```
sudo apt install cpufrequtils
```

по умолчанию после инсталляции ОС использует режим "ondemand" - "по запросу" (производительность CPU по запросу приложений, экономит электроэнергию , но ниже производительность):

```
cat /etc/init.d/cpufrequtils | grep GOVERNOR=
```

в выходном сообщении системы , режим работы по умолчанию после инсталляции - "ondemand" :

```
GOVERNOR="ondemand"
```

Установить режим - результативность/производительность - в файле /etc/init.d/cpufrequtils значение "**ondemand**" заменить на "**performance**"

```
sudo sed -i 's/GOVERNOR="ondemand"/GOVERNOR="performance"/g' /etc/init.d/cpufrequtils
```

Перезапустить утилиту:

```
sudo /etc/init.d/cpufrequtils restart
```

Затем выполнить команду:

```
sudo systemctl daemon-reload
```

Отключить SWAP

Ubuntu сервер SSW работает **в реальном масштабе времени** ,
поэтому **все** необходимые **данные должны находится в оперативной памяти,** использование файла подкачки (swap-файл - /swap. img) может привести к увеличению времени обработки вызовов приложения SSW ECSS10 , **что недопустимо**. Swap - отключаем.
Выполнить последовательно три команды:

Отключение Swap:

```
sudo swapoff -a
```

Удалить файл swap.img.

```
sudo rm /swapfile
```

Удалить эту строчку (//swapfile swap swap defaults 0 0)

```
sudo nano /etc/fstab
```

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
```

```
# / was on /dev/sda2 during curtin installation
UUID=4c88cb4a-42c9-4b52-98c2-78da7e23b621 /                       ext4    defaults
1 1
```

Для проверки выполните команду **free -h:**

```
free -h
```

Размер Swap равен 0 - то есть он выключен

```
free -h
              total        used        free      shared  buff/cache   available
Mem:           3,9G        110M        3,2G        820K        535M        3,5G
Swap:            0B          0B          0B
```

Установка часового пояса

При инсталляции astra linux **не предлагается** установить часовой пояс(по умолчанию устанавливается время UTC). Его нужно установить **вручную** (для корректной работы системы тарификации , работ по расписанию и т.д.), например:

```
sudo timedatectl set-timezone Asia/Novosibirsk
```

Улучшение работы высоконагруженных серверов

Улучшить работу высоконагруженных серверов можно увеличив лимит открытых файлов.

Для установки лимита открытых файлов необходимо:

Проверить текущий лимит командой:

```
ulimit -a
```

Результат :

```
eltex@ecss1:~$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
scheduling priority             (-e) 0
file size               (blocks, -f) unlimited
pending signals                 (-i) 15515
max locked memory       (kbytes, -l) 65536
max memory size         (kbytes, -m) unlimited
open files                      (-n) 1024
pipe size            (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
real-time priority              (-r) 0
stack size              (kbytes, -s) 8192
```

```
cpu time               (seconds, -t) unlimited
max user processes              (-u) 15515
virtual memory          (kbytes, -v) unlimited
file locks                      (-x) unlimited
```

Данного лимита (open files **1024**) **недостаточно** для нормальной работы высоконагруженных серверов.

Установить лимит открытых файл для каждого пользователя:

```
sudo sed -i  '55i\*               soft    nproc           65536\n*                hard
nproc           131072\n*               soft    nofile          65536\n*                hard
nofile          131072\nroot            -       memlock         unlimited' /etc/security/
limits.conf
```

Результат:

```
# /etc/security/limits.conf
#
#Each line describes a limit for a user in the form:
#
#<domain>        <type>  <item>  <value>
#
#Where:
#<domain> can be:
#        - a user name
#        - a group name, with @group syntax
#        - the wildcard *, for default entry
#        - the wildcard %, can be also used with %group syntax,
#                 for maxlogin limit
#        - NOTE: group and wildcard limits are not applied to root.
#          To apply a limit to the root user, <domain> must be
#          the literal username root.
#
#<type> can have the two values:
#        - "soft" for enforcing the soft limits
#        - "hard" for enforcing hard limits
#
#<item> can be one of the following:
#        - core - limits the core file size (KB)
#        - data - max data size (KB)
#        - fsize - maximum filesize (KB)
```

```
#        - memlock - max locked-in-memory address space (KB)
#        - nofile - max number of open files
#        - rss - max resident set size (KB)
#        - stack - max stack size (KB)
#        - cpu - max CPU time (MIN)
#        - nproc - max number of processes
#        - as - address space limit (KB)
#        - maxlogins - max number of logins for this user
#        - maxsyslogins - max number of logins on the system
#        - priority - the priority to run user process with
#        - locks - max number of file locks the user can hold
#        - sigpending - max number of pending signals
#        - msgqueue - max memory used by POSIX message queues (bytes)
#        - nice - max nice priority allowed to raise to values: [-20, 19]
#        - rtprio - max realtime priority
#        - chroot - change root to directory (Debian-specific)
#
#<domain>      <type>  <item>         <value>
#

#*               soft    core            0
#root            hard    core            100000
#*               hard    rss             10000
#@student        hard    nproc           20
#@faculty        soft    nproc           20
#@faculty        hard    nproc           50
#ftp             hard    nproc           0
#ftp             -       chroot          /ftp
#@student        -       maxlogins       4
*                soft    nproc           65536
*                hard    nproc           131072
*                soft    nofile          65536
*                hard    nofile          131072
root             -       memlock         unlimited

# End of file
```

13.3 Настройка сетевых интерфейсов

⚠ Получение адресов на сетевых интерфейсах по DHCP **недопустимо** на серверах ECSS.

Сетевые настройки необходимо выполнять с помощью Netplan.

По умолчанию в Astra Linux пакет netplan не установлен.

Установить пакеты для обслуживание функции netplan:

```
sudo apt install -y dialog libssl1.0.0 netplan.io
```

Сейчас можно выполнить конфигурацию netplan:

```
sudo nano /etc/netplan/ecss.yaml
```

13.3.1 Пример netplan

Netplan для интерфейсов сервера ecss1 (/etc/netplan/ecss.yaml)

```
# Netplan для хоста ecss1 программного коммутатора
# Обратите внимание на обязательное наличие минимум двух пробелов в каждой строке и секции
 (кроме строки секции network)

network:
  version: 2 # Версия netplan
  renderer: networkd # Исполнитель конфигураций netplan
  ethernets: # Раздел описания интерфейсов ethernet
    enp0s3: # Название интерфейса ВМ к интернет
        dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
        dhcp6: no
        addresses: [192.168.56.40/24]
    enp0s8: # Название интерфейса для SSW
        dhcp4: no # Отключаем на интерфейсах динамическое распределение IP-адреса
        dhcp6: no
  vlans:
    net.10: # Интерфейс управления
        id: 10
        link: enp0s3
        addresses: [10.0.10.40/24]
    net.20: # Интерфейс для VoIP
        id: 20
        link: enp0s8
        addresses: [10.0.20.40/24]
```

Так же проверить отсутствие в каталоге **/etc/netplan/** других файлов, если другие файлы присутствуют , то их нужно переместить в другой каталог или удалить, в противном случае возможна некорректная настройка сетевых интерфейсов и некорректная работа SSW.

```
ls -la /etc/netplan/
```

Применить, установленные параметры командой:

```
sudo netplan apply
```

Просмотреть получившиеся настройки можно при помощи команд **sudo ifconfig** или **ip a**:

Для применения новых сетевых настроек необходимо выполнить команду **netplan apply.** Перезапуск сети или системы не требуется. Подробнее про настройки netplan см. в Приложении E. Netplan.

13.4 Инсталляция необходимого ПО

13.4.1 Установка дополнительных пакетов программ

Для работы SSW, потребуются дополнительные пакеты ПО не установленные в Astra Linux по умолчанию, но доступные в репозитории Eltex.

```
sudo apt install -y ntp iptables tcpdump vlan dnsmasq aptitude atop ethtool htop mc screen ssh
 tftpd-hpa sngrep tshark gnuplot debconf-utils ncdu wget
```

Пакеты ПО которые рекомендуется установить для работы

Список **обязательного** сервисного программного обеспечения:

```
sudo apt install ntp tcpdump vlan dnsmasq
```

ntp	NTP-сервер
tcpdump	снифффер пакетов
vlan	управление VLAN
dnsmasq	легковесный DNS/DHCP-сервер

Список **рекомендуемого** диагностического и вспомогательного программного обеспечения:

```
sudo apt install aptitude atop ethtool htop mc screen ssh tftpd sngrep tshark gnuplot
debconf-utils iotop ncdu
```

aptitude	установка программ из репозиториев, рекомендуется использовать вместо программы apt/apt-get

atop	мониторинг загрузки хоста с функцией периодического сохранения информации в файлы
ethtool	просмотр статистики сетевых интерфейсов
htop	мониторинг процессов
mc	файловый менеджер
screen	мультиплексор терминалов
ssh	сервер и клиент SSH
tftpd	TFTP-сервер
sngrep	трассировка sip
tshark	консольный аналог wireshark
gnuplot	вывод графиков статистики
debconf-utils	набор утилит для работы с базой debconf
iotop	инструмент для мониторинга использования ввода-вывода (IO) на диске в реальном времени в Linux
ncdu	утилита для поиска больших директорий в системе Linux

Для **просмотра** установленных пакетов выполнить следующую команду (**данный пункт является необязательным**: его можно выполнить, если не уверены, что какие-то пакеты установлены):

```
sudo dpkg --get-selections
```

⚠ Перед началом установки пакетов ecss , нужно убедится в соответствии полосы пропускания СПД необходимым требованиям.
Для этого выполнить команду **sudo ethtool <имя интерфейса>** для всех физических интерфейсов.
Пример:

```
sudo ethtool net.20
Settings for net.20:
    Supported ports: [ TP ]
    Supported link modes:   10baseT/Half 10baseT/Full
                            100baseT/Half 100baseT/Full
                            1000baseT/Full
    Supported pause frame use: No
    Supports auto-negotiation: Yes
    Supported FEC modes: Not reported
    Advertised link modes:  10baseT/Half 10baseT/Full
                            100baseT/Half 100baseT/Full
                            1000baseT/Full
    Advertised pause frame use: No
    Advertised auto-negotiation: Yes
    Advertised FEC modes: Not reported
    Speed: 1000Mb/s
    Duplex: Full
    Auto-negotiation: on
    Port: Twisted Pair
    PHYAD: 0
    Transceiver: internal
    MDI-X: off (auto)
    Link detected: yes
```

Проверить значение следующих параметров:
Advertised auto-negotiation: **Yes**
Speed: **1000Mb/s** (не менее)
Duplex: **Full**

13.5 Инсталляция пакетов ECSS

13.5.1 Предварительные требования

- Установленная и обновленная операционная система Astra Linux;
- Отсутствие в системе пользователя с именем **ssw**;
- Разбиение дискового пространства в соответствии с рекомендациями;
- Настроенная сеть;
- Установленный набор необходимых пакетов;

- Наличие локального файла репозитория Eltex.

установка пакетов deb

```
   ┌ версия ПО ECSS. Сейчас в заявке на сертификацию она заявлена как 3.

   │ Мы её менять не будем до следующей сертификации или
  │ выпуска принципиально новой версии системы.

   │   ┌ версия System Release. Общая версия для всех компонент входящих в конкретный релиз.

   │      │ Меняем её централизовано,
   │   │ принимаем решение о выпуске нового релиза. Релизы между собой обычно не совместимы.
┌──┘ ┌──┘
ECSS.SysRel.SubMaj.SubMin
              └────┘ └────┘
        │   └ версия Subsystem minor. Минорную версию подсистемы устанавливает разработчик
подсистемы.

                  │          Минорную версию
        │     меняем при добавлении патча. Как правило, минорные версии в рамках одной мажорной
версии
        │     совместимы между собой и отличаются в рамках конкретных патчей.

                  └ версия Subsystem major. Мажорная версия устанавливается разработчиком
подсистемы.

                     Мажорную версию подсистемы необходимо менять при внесении в подсистему
существенных изменений.
```

Утилита установки пакетов APT анализирует версию пакета слева направо, для примера, мы имеем пакет 14.14.7.7 ,
в репозитории находятся пакеты
14.14.7.8
14.14.7.9
14.14.8.1
14.14.20 -14.14.28
то при выполнении команды **sudo apt install имя пакета** → будет автоматически проверен и установлен пакет 14.14.28 , так как он самый последний, анализ будет выполнен по 3-й позиции (major), анализ по 4-й (minor) выполнен не будет (аналогичным образом будет выполнятся команда **sudo apt upgrade --enable-upgrade**).
В случае если для конкретной ситуации требуется перейти с версии 14.14.7.7 на версию 14.14.7.9 , стандартная команда → **sudo apt upgrade --enable-upgrade** , нам не поможет, так как будет выбран самый новый пакет, в данной ситуации нам **необходимо в явном виде** указать какую версию пакета мы хотим установить, в данном примере мы **должны** выбрать команду → **sudo apt install имя пакета=14.14.7.9** . Обычно это необходимо для тестирования определенного патча, для стандартных обновлений достаточно выбора привычной команды установки/обновления пакета

В ходе установки пакетов ECSS нужно будет ответить на ряд вопросов для формирования необходимой конфигурации. Для автоматической загрузки требуемых настроек можно воспользоваться командами из пакета debconf-utils.
Описание и примеры использования при работе с debconf приведены в Приложении B. Debconf.

Для инсталляции системы ECSS-10 необходимо устанавливать пакеты в порядке, в котором они описаны ниже в документации.

13.5.2 Инсталляция обязательных пакетов

Установка пакета ecss-dns-env

Выполнить следующую команду:

на обоих хостах

```
sudo apt install -y ecss-dns-env
```

Настройщик предложит выбрать разделы для настройки по вопросам ниже. **Не выбираем ничего** просто нажать **Enter**.

Настройка преобразование DNS имени в IP адрес

Для корректной работы системы, после установки пакета ecss-node требуется скорректировать дефолтные установки . Для этого:

В файле /etc/resolv.conf изменить параметр nameserver установленный системой по умолчанию на 127.0.0.1 (если строк nameserver несколько - остальные удалить/закомментировать):

```
sudo nano /etc/resolv.conf
```

Пример:

```
domain eltex.loc
search eltex.loc
nameserver 127.0.0.1
```

В Astra Linux после перезагрузки файл /etc/resolv.conf приводится к исходному виду, поэтому его **нужно сделать неизменяемым**:

```
sudo chattr +i /etc/resolv.conf
```

Для доступа к репозиториям AstraLinux и Элтекс нужно указать локальный DNS или глобальный (для примера 8.8.8.8 или 77.88.8.8)

Добавить в файл /etc/dnsmasq.d/ecss строки server=<Ipadd DNS сервера> (в примере 77.88.8.8) :

```
sudo sh -c "echo 'server=77.88.8.8' >> /etc/dnsmasq.d/ecss"
```

Перезапустить сервис:

```
sudo systemctl restart dnsmasq.service
```

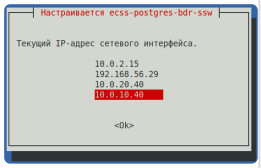
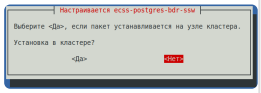
Установка Postgres

Установка пакета ecss-postgres-bdr-ssw

ecss1

```
sudo apt install -y ecss-postgres-bdr-ssw
```

При конфигурации пакета **ecss-postgres-bdr-ssw** будут заданы вопросы:

Вопросы ecss-postgres-bdr-ssw	Ответы для ecss1 (Master)	Пример
Текущий IP-адрес сетевого интерфейса.	**Выбрать** из списка IP-адрес для работы с Postgres	
Установка в кластере?	**No** (необходимо выбрать)	

Результат:

```
============================================================
============================================================
 ecss-postgres-bdr-ssw successfully installed/updated!
============================================================
============================================================
```

Установка ecss-node

Установка обязательного пакета **ecss-node** включает в себя установку и первоначальную настройку основных подсистем.

⚠ В системе уже должен быть установлен пакет **ecss-postgres-bdr-ssw**

Для установки пакета **ecss-node** выполните команду:

первоначально на ecss1

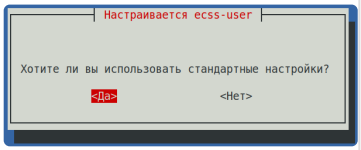
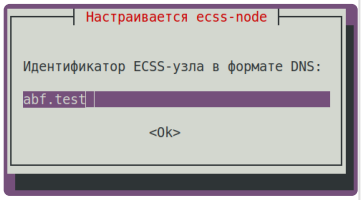
```
sudo apt install -y ecss-node
```

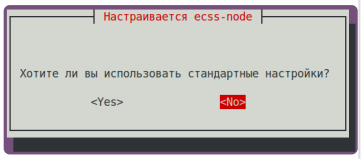
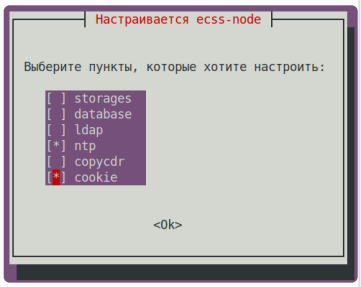
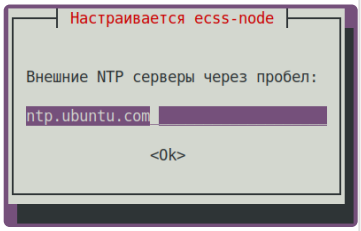
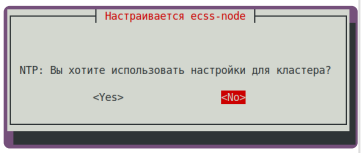
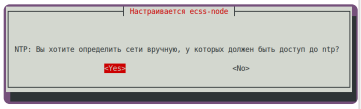
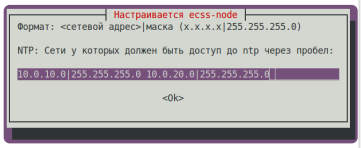
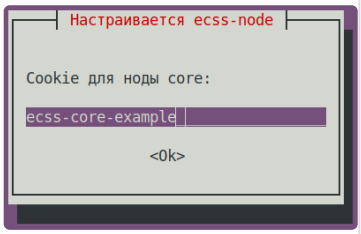
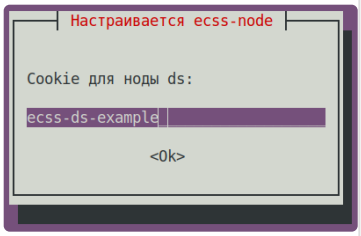
Во время установки пакета создается пользователь **ssw**, от имени которого запускаются все сервисы ecss*. Создаются необходимые каталоги, выполняется настройка DNS, идет настройка SSL-сертификатов.

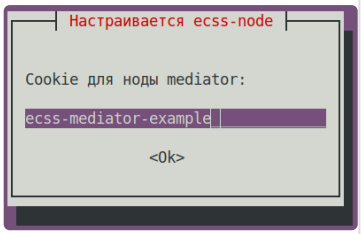
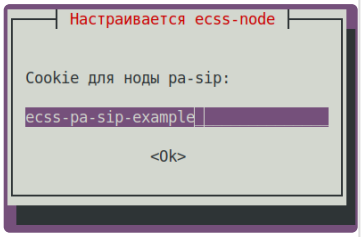
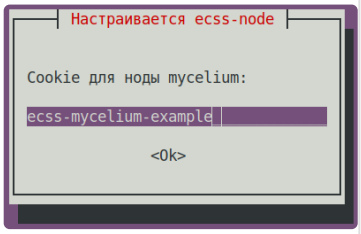
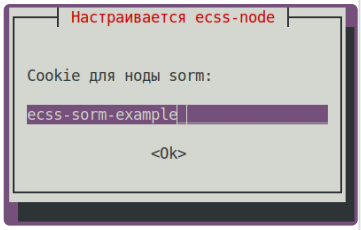
✓ При установки так же будет установлен пакет **ecss-user**

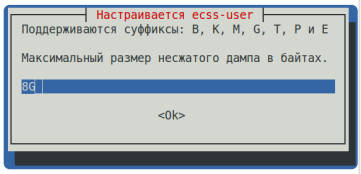
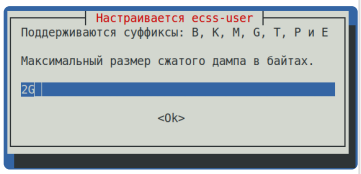
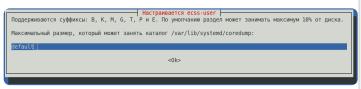
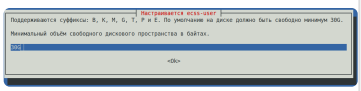
Установка на сервере **ecss1**

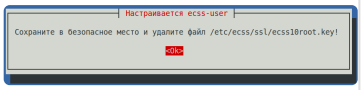
Во время установки будет предложено настроить параметры, необходимые для формирования конфигурационных файлов, примеры ответов следует ниже.

Вопросы ecss-user	**Ответы для ecss1**	**Пример**
Хотите ли вы использовать стандартные настройки?	Yes (значение по умолчанию)	
Вопросы ecss-node	**Ответы для ecss1**	**Пример**
Идентификатор ECSS-узла в формате DNS (Настройка имени кластера): (любое уникальное имя, в примере abf.test)	**abf.test** (необходимо ввести) ✓ Формат **ECSS_ID** Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире. `[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]`	

Хотите ли вы использовать стандартные настройки?	**No** (необходимо выбрать)	
Выберите пункты, которые хотите настроить:	**ntp + cookie** (необходимо выбрать)	
Внешние NTP серверы через пробел:	ntp.ubuntu.com (значение по умолчанию)	
NTP: Вы хотите использовать настройки для кластера?	**No** (необходимо выбрать)	
NTP: Вы хотите определить сети вручную, у которых должен быть доступ до ntp?	Yes (значение по умолчанию)	
NTP: Сети у которых должен быть доступ до ntp через пробел: (в примере 10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0)	**10.0.10.0|255.255.255.0 10.0.20.0|255.255.255.0** (необходимо ввести)	
Cookie для ноды core: (указать уникальный cookie для core , в примере ecss-core-example)	**ecss-core-example** (необходимо ввести)	
Cookie для ноды ds: (указать уникальный cookie для ds , в примере ecss-ds-example)	**ecss-ds-example** (необходимо ввести)	

Cookie для ноды mediator: (указать уникальный cookie для mediator , в примере ecss-mediator-example)	**ecss-mediator-example** (необходимо ввести)	
Cookie для ноды pa-sip: (указать уникальный cookie для pa-sip , в примере ecss-pa-sip-example)	**ecss-pa-sip-example** (необходимо ввести)	
Cookie для ноды mycelium: (указать уникальный cookie для mycelium , в примере ecss-mycelium-example)	**ecss-mycelium-example** (необходимо ввести)	
Cookie для ноды sorm: (указать уникальный cookie для sorm , в примере ecss-sorm-example)	**ecss-sorm-example** (необходимо ввести)	

Вопросы ecss-user	**Ответы для ecss1**	**Пример**
Максимальный размер несжатого дампа в байтах.	8G (значение по умолчанию)	
Максимальный размер сжатого дампа в байтах.	2G (значение по умолчанию)	
Максимальный размер, который может занять каталог /var/lib/systemd/coredump:	default (значение по умолчанию)	
Минимальный объем свободного дискового пространства в байтах.	30G (значение по умолчанию)	

Сохраните в безопасное место и удалите файл / etc/ecss/ssl/ecss10root.key!	Ok (значение по умолчанию)	

Проверить статус сервисов следующей командой:

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
```

В случае статуса "active" продолжаем дальше, в случае статуса "inactive" выполнить команду "sudo systemctl restart ecss-<имя пакета>"

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
active
active
active
active
active
```

Установка лицензии

Система ECSS-10 использует ELM-лицензирование – сервис распределения лицензий ECSS-экосистемы, состоит из **ecss-license-agent** и **ecss-license-master** .

- **ECSS License Agent** - Адаптер для работы с EcssLicenseMaster (LM). Получает список хостов EcssLicenseMaster, реализует логику подключения и переподключения к LM.
  - Отвечает за получение, валидацию и дальнейшую отправку лицензий клиенту.
  - Отправляет статусы подключения и полученные лицензии клиенту.
  - Предоставляет API для отправки HTTP запросов и WS событий на LM.
- **Ecss License Master** - Сервис оркестрации параметров лицензии для ECSS-узлов.
  - Отвечает за:
    - Получение лицензии для конкретного ECSS-узла (SSW / AuP) либо устройства (SMG)
    - Распределение параметров лицензии между ECSS-узлами и ECSS-сервисами
    - Контроль активности ECSS-узлов и их валидация во избежание тиражирования лицензии сверх указанных лимитов
    - Контроль изменения указанных лимитов на ECSS-узлах, и их динамическое распределение в соответствии с текущей нагрузкой.

ecss-license-agent является составной частью SSW, входит в пакет ecss-node. Отдельной конфигурации не требует.

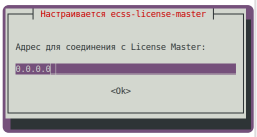
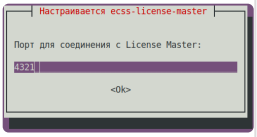
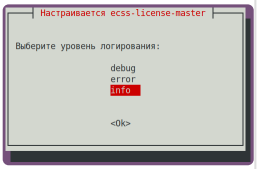
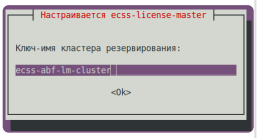
В свою очередь **Ecss License Master** может быть установлен как на SSW так и на отдельно выделенный сервер. В случае если Ecss License Master уже установлен на вашем проекте, то потребуется только дополнить запись о новом компоненте в конфигурационном файле LM, и следующий пункт – установка "Установка LM" не потребуется.

Установка LM на SSW host

Установить пакет LM (ecss-license-master)

```
sudo apt install ecss-license-master
```

При установке пакета ecss-license-master будут заданы вопросы:

Вопросы	Ответы	Примеры
Адрес для соединения с License Master: License Master listen ip:	0.0.0.0 (значение по умолчанию)	
Порт для соединения с License Master: License Master listen port:	4321 (значение по умолчанию)	
Выберите уровень логирования: Log level:	info (значение по умолчанию), возможные варианты: debug, error, info	
Ключ-имя кластера резервирования: License Master cluster key:	указать уникальное имя ecss-lm-cluster (значение по умолчанию) Менеджеры с одинаковыми именами попытаются собраться в кластер и резервировать друг друга.	

Данные сохраняются в файле - **/etc/ecss/ecss-license-master/config.env** :

```
ECSS_LM_HOST=0.0.0.0
ECSS_LM_PORT=4321
ECSS_LM_LOG_LEVEL=info
RELEASE_COOKIE=ecss-abf-lm-cluster
```

Конфигурация LM

Сервис ecss-license-master использует два конфигурационных файла: **/etc/ecss/ecss-license-master/config.env** и **/etc/ecss/ecss-license-master/config.yaml**.
В файл **config.env** информация сохраняется в момент установки пакета **ecss-license-master**.
Config.yaml файл содержит основные настройки для подключения к ELM серверу, откуда брать лицензии, какие лицензии, и куда их потом передавать, **все это нужно сконфигурировать вручную**.

При установке LM на SSW

Выполнить команду:

```
sudo nano /etc/ecss/ecss-license-master/config.yaml
```

информация в файле после установки

```
elm_addresses: []
licenses: []
ecss_nodes: []
```

В поле **elm_adresses** нужно прописать список серверов ELM к которым мы хотим подключаться для получения лицензий. Первый адрес в списке используется как основной. Дополнительные строки с адресами работают как резерв.

Данные ELM сервера(ов) получить у менеджера проекта.

пример:
- "elm-1.eltex.loc:8099"
- "192.168.111.22:8099"
- "elm-3.eltex.loc:8099"

Дальше задаются сами лицензии в поле **licenses**.

- **id** – может быть любой, главное чтоб был уникальный у каждой лицензии. Влияет только на адрес лицензии на самом LM (например 0,1,2,3).
- **kind** – для SSW поддерживается только ***ssw*.**
- **type** – выбирается как ***elm*** если запрашивать данную лицензию нужно с ELM сервера. Если нужно использовать файловую лицензию SSW, то тип выбирается как ***ecss_license***.
  - при ***elm*** типе, необходимо задать поля **license_key** и **product_id** для идентификации лицензии.
  - при ***ecss_license*** типе, задается поле **license** содержащее ключ статичной лицензии

Параметры **license_key** и **product_id** должны быть согласованы с данными загруженными на ELM сервере .

Значение параметров **license_key**, **product_id** или **license**, **passport** нужно получить у менеджера проекта.

Блок **ecss_nodes**.
В нем нужно задать данные сервисов которые будут получать лицензию. В нашем случае SSW.
Сервис задается тремя параметрами:

- **id –** идентификатор сервиса, должен совпадать с ECSS_ID лицензируемого SSW.
- **kind –** тип лицензируемого сервиса. по аналогии с блоком лицензий, для SSW поддерживается только ***ssw*.**
- **license_id –** идентификатор лицензии которая будет отправляться на данный сервис. Соответствует идентификаторам лицензии из блока **licenses**

Два SSW с **разными ECSS_ID** не могут получить одну и ту же лицензию, конфигурация с таким распределением будет помечена некорректной.
Кроме того, два SSW с **одинаковыми ECSS_ID** не смогут одновременно получать одну и ту же лицензию с одного LM,
поэтому крайне важно корректно устанавливать свой ECSS_ID и делать его уникальным как минимум в зоне видимости используемого LM.

Формат **ECSS_ID**
Начинается и заканчивается обязательно латинской буквой или цифрой, в теле же могут использоваться точки и тире.

```
[A-Za-z0-9][A-Za-z0-9.-][A-Za-z0-9]
```

Для SSW работающего в кластере указываем только одну запись в блок **ecss_nodes** на кластер, индивидуально для каждого хоста указывать нельзя. Иначе будут генерироваться два запроса с одинаковым ECSS_ID что недопустимо.

Пример: файла config.yaml

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
   kind: SSW
   type: elm
   license_key: "ssw1test"
   product_id: "ECSS1000001"
 - id: 1
   kind: ssw
   type: ecss_license
   license: "af615ebb92d381125ff"
ecss_nodes:
 - id: abf.test
   kind: ssw
   license_id: 0
```

Выполнить перезагрузку сервиса **ecss-license-master.service** для обновления конфигурации из файла **/etc/ecss/ecss-license-master/config.yaml** следующей командой:

если кластер, то на обоих хостах

```
sudo systemctl restart ecss-license-master.service
```

При подключении SSW к ранее установленному LM

В случае если LM уже установлен и используется в локальной сети требуется добавить параметры нового SSW в конфигурационный файл **/etc/ecss/ecss-license-master/config.yaml** в блоках licenses и ecss_nodes.

Выполнить команду:

если кластер, то на обоих хостах

```
sudo nano /etc/ecss/ecss-license-master/config.yaml
```

для примера:

Было:

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
   kind: SSW
   type: elm
   license_key: "ssw1test"
   product_id: "ECSS1000001"
 - id: 1
   kind: ssw
   type: ecss_license
   license: "af615ebb92d381125ff"
 - id: 2
   kind: SSW
   type: elm
   license_key: "ssw2test"
   product_id: "ECSS1000002"
ecss_nodes:
 - id: 1.test
   kind: ssw
   license_id: 0
 - id: 2.test
   kind: ssw
   license_id: 1
 - id: 3.test
   kind: ssw
   license_id: 2
```

Стало:

```
elm_addresses:
 - "elm-1.eltex.loc:8099"
 - "192.168.111.22:8099"
 - "elm-3.eltex.loc:8099"
licenses:
 - id: 0
```

```
    kind: SSW
    type: elm
    license_key: "ssw1test"
    product_id: "ECSS1000001"
  - id: 1
    kind: ssw
    type: ecss_license
    license: "af615ebb92d381125ff"
  - id: 2
    kind: SSW
    type: elm
    license_key: "ssw2test"
    product_id: "ECSS1000002"
  - id: 3
    kind: SSW
    type: elm
    license_key: "ssw3test"
    product_id: "ECSS1000003"
ecss_nodes:
  - id: 1.test
    kind: ssw
    license_id: 0
  - id: 2.test
    kind: ssw
    license_id: 1
  - id: 3.test
    kind: ssw
    license_id: 2
  - id: abf.test
    kind: ssw
    license_id: 3
```

Выполнить перезагрузку сервиса **ecss-license-master.service** для обновления конфигурации из файла **/etc/ecss/ecss-license-master/config.yaml** следующей командой:

```
sudo systemctl restart ecss-license-master.service
```

Подключение SSW к Менеджера Лицензий

Выполнить конфигурацию подключения SSW к LM следующей командой в CoCon, (в примере LM , IPadd 10.0.10.41. значение порта из файла /etc/ecss/ecss-license-master/config.env по умолчанию 4321):

выполняется в CoCon

```
/system/licence/manager/set --hosts [https://10.0.10.41:4321]
```

В случае установки LM на SSW указывать IP адреса OAM

После выполнения проверить статус подключения командой:

выполняется в CoCon

```
/system/licence/manager/show-status
┌──────────────────────────┬───────┬─────┐
│          Host            │Current│Alive│
├──────────────────────────┼───────┼─────┤
│https://10.0.10.41:4321   │*      │true │
└──────────────────────────┴───────┴─────┘
```

Должны иметь статус alive=true, и быть "current".

- Current - отображает к какому хосту подключён SSW. Если есть подключение, указывается *, если нет - ничего не указано.
- Alive - отображает статус доступности хоста (healthcheck).

Подключение к License Manager

Отсутствие подключения к License Manager \ ELM будет равноценно отсутствию доступа к токену, что приведёт к критическим авариям, переходу в аварийный режим по истечению которого, если связь не будет восстановлена, система перейдёт на дефолтную лицензию.

данное поведение применимо только при работе в **elm** режиме

Если статус к LM показан current, можно посылать запрос на загрузку лицензии на SSW командой:

выполняется в CoCon

```
/cluster/storage/ds1/licence/request
```

пример выполнения команды /cluster/storage/ds1/licence/request

type=elm

```
/cluster/storage/ds1/licence/request
Licence received
[*******                                                    ] 6s 2ms
Success: Licence parameters applied
```

В режиме "type=ecss_license" команда **/cluster/storage/ds1/licence/request** выполнится только в случае если был предварительно загружен паспорт.

type=ecss_license

```
/cluster/storage/ds1/licence/request
Waiting for licence...
[*******                                                    ] 6s
┌─┬──────────────────────────────────────────────────────┬─────────┬──────────────────────────────
──┐
│A│                 Description                          │Old Value│              New Value
  │
```

```
|-|-----------------------------------------------------------|---------|-------------------------
-----------|
|^|Support for reservation of call-processes                  |false    |true
           |
|^|Maximum call duration (in seconds)                         |60       |2678400
           |
|^|Maximum number of simultaneous calls                       |5        |10000
           |
|^|Elph                                                       |         |
           |
|^| total count of members of the one Elph group              |10       |100
           |
|^|Subscribers limit                                          |10       |500000
           |
|^|Virtual subscribers limit                                  |infinity |1000
           |
|^|Add-on conferences                                         |         |
           |
|^| total count                                               |30       |1000
           |
|^|Add-on conferences                                         |         |
           |
|^| members of the one add-on conference                      |16       |300
           |
|^|Chat rooms                                                 |         |
           |
|^| total count                                               |30       |100
           |
|^|Chat rooms                                                 |         |
           |
|^| members of the one chatroom                               |16       |300
           |
|^|Meet Me                                                    |         |
           |
|^| total count                                               |0        |100
           |
|^|Meet Me                                                    |         |
           |
|^| members of the one Meet Me conference                     |0        |300
           |
|^|SORM                                                       |         |
           |
|^| enabled                                                   |false    |true
           |
|+| channels on SORM mediator                                 |         |64
           |
|^|Sorm extractor                                             |none     |mfi
           |
|^|Support antifraud system                                   |none     |custom
           |
|+|Call center                                                |         |
           |
|+| operator's capability for look at calls in queue          |         |true
           |
|+| operator's capability for use Intervension SS             |         |true
           |
|^|Call center                                                |         |
           |
```

```
|^| active agents                                          |0        |1000
              |
|^|Call center                                            |         |
              |
|^| active supervisors                                     |0        |100
              |
|^|Teleconference                                          |         |
              |
|^| channels                                               |3        |2000
              |
|^|Teleconference                                          |         |
              |
|^| max members of one teleconference                      |0        |200
              |
|^|Teleconference                                          |         |
              |
|^| active count                                           |0        |32
              |
|^|TSMN system                                             |         |
              |
|^| concurrent calls (active)                              |0        |50
              |
|^|TSMN system                                             |         |
              |
|^| concurrent calls (backup)                              |0        |50
              |
|^|TSMN system                                             |         |
              |
|^| backup mode                                            |none     |"backup"
              |
|^|Total count of simultaneous records voice calls         |0        |200
              |
|+|IVR                                                     |         |
              |
|+| customization enabled                                  |         |true
              |
|+| Automatic Speech Recognition (ASR) subsystem enabled|         |true
              |
|^|Channels on dialer outgoing calls                       |2        |5
              |
|+|Supplementary Services licence package(s)               |         |
              |
|+| name                                                   |         |'ECSS-ADV'
              |
|+| limit                                                  |         |10000
              |
|+| description                                            |         |"Additional services of 4
and 5 levels" |
|+| SS list                                                |         |[1,2,3,4,5]
              |
|+| name                                                   |         |'ECSS-BAS'
              |
|+| limit                                                  |         |10000
              |
|+| description                                            |         |"Basic services of 1 and
2 levels"      |
|+| SS list                                                |         |[1,2]
              |
```

```
|+| name                                                    |         |'ECSS-BAS+'
              |
|+| limit                                                   |         |10000
              |
|+| description                                             |         |"Expansion of basic
services of 3 level"|
|+| SS list                                                 |         |[1,2,3]
              |
|^|GEO backup                                              |         |
              |
|^| enabled                                                 |false    |true
              |
|^|SIGTRAN                                                  |         |
              |
|^| SEP                                                     |false    |true
              |
|^| STP                                                     |false    |true
              |
└─┴─────────────────────────────────────────────────────────┴─────────┴──────────────────────
──────────────┘
Legend:
    '^' - Changed;
    '-' - Deleted;
    '+' - New.


[request] You are trying to add a license that is different from the current one by the pa
          rameters in the table (other values will remain unchanged).
Licence received      o ?> yes
[*******                                                              ] 1m 15s
Success: Licence parameters applied
```

В зависимости от выбранного типа лицензии для SSW : "type=**elm**" или "type=**ecss_license**" на команду просмотра лицензий получим разный вывод:

type=ecss_license

```
/cluster/storage/ds1/licence/list-licence
┌─┬──────────────────┬──────────┬──────┬──────────────────┬──────────────────────┬──────────
──────────┬──────────┐
|Id|Creation date(UTC) |   SSW ID   |Active|  Description   |Comm. Exp.
Date(UTC)|Expiration date(UTC)|Time left|
├─┼──────────────────┼──────────┼──────┼──────────────────┼──────────────────────┼──────────
──────────┼──────────┤
|1 |12.03.2026 18:40:00|ECSS 010070 |*     |ECSS TPM License|13.03.2027 06:59:59
|14.03.2026 10:44:05 |1d 6h 59m|
|0 |01.01.1990 00:00:00|ECSS DEFAULT|      |Default licence |                      |
|          |
└─┴──────────────────┴──────────┴──────┴──────────────────┴──────────────────────┴──────────
──────────┴──────────┘
```

TPM - является полным аналогом лицензирования которое использовалось раньше, с токеном и паспортом, без подключения к LM.

или

type=elm

```
/cluster/storage/ds1/licence/list-licence
┌──┬────────────────────┬─────────────┬──────┬─────────────────┬────────────────────┬──────────
──────────────────┬──────────┐
|Id|Creation date(UTC) |   SSW ID    |Active|  Description    |Comm. Exp.
Date(UTC)|Expiration date(UTC)|Time left|
├──┼────────────────────┼─────────────┼──────┼─────────────────┼────────────────────┼──────────
──────────────────┼──────────┤
|1 |03.02.2026 16:26:34|ECSS2000009 |*     |ECSS ELM License|03.02.2027 16:26:34
|14.03.2026 10:49:12 |1d 6h 59m|
|0 |01.01.1990 00:00:00|ECSS DEFAULT|      |Default licence |                    |
|          |
└──┴────────────────────┴─────────────┴──────┴─────────────────┴────────────────────┴──────────
──────────────────┴──────────┘
```

Так же требуется выполнить конфигурацию для Sip адаптера , выполнив следующую команду:

на обоих хостах

```
sudo systemctl edit ecss-pa-sip.service
```

Добавить в конфигурационный файл следующие данные:

```
[Service]
LimitNOFILE=65536
```

Перезагрузите конфигурацию

на обоих хостах

```
sudo systemctl daemon-reload
```

Актуализировать конфигурационный файл, выполнив перезагрузку сервера:

```
sudo reboot
```

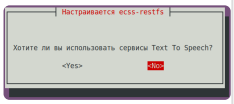
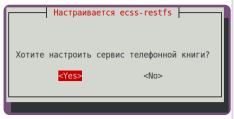
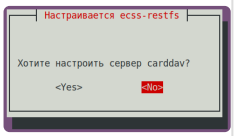
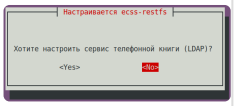
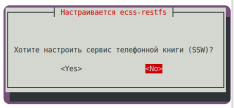
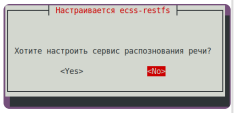
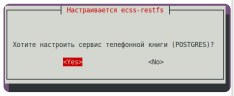
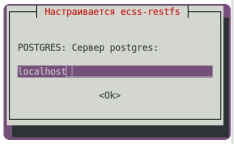
Установка остальных пакетов ecss и их настройка

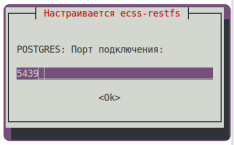
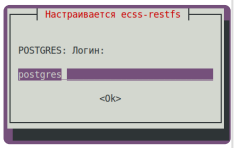
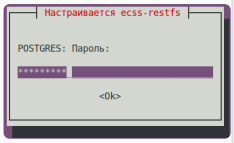
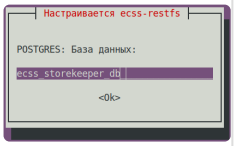
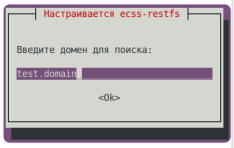
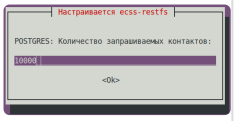
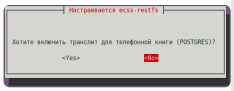
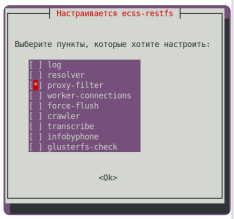
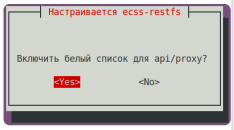
Установка пакета ecss-restfs и его настройка

установить на ecss1, затем на ecss2

```
sudo apt install -y ecss-restfs
```

Установка **ecss-restfs.** В ходе установки нужно будет ответить на ряд вопросов для создания необходимых конфигурационных файлов. Также инсталлятор предложит установить и настроить пакет Text2speech от Yandex.

Вопросы ecss-restfs	Ответы	Пример
Хотите ли вы использование функции Text To Speeh (Use TTS service)	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (Configure phone book)	**Yes** (нужно указать)	
Хотите настроить сервер carddav?	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (LDAP)?	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (SSW)?	No (значение по умолчанию)	
Хотите настроить сервис распознавания речи (Configure speech recognition service)	No (значение по умолчанию)	
Хотите настроить сервис телефонной книги (POSTGRES)?	Yes (значение по умолчанию)	
POSTGRES: Сервер postgres:	localhost (значение по умолчанию)	

Вопросы ecss-restfs	Ответы	Пример
POSTGRES: Порт подключения:	5439 (значение по умолчанию)	
POSTGRES: Логин:	postgres (значение по умолчанию)	
POSTGRES: Пароль:	postgres1 (значение по умолчанию)	
POSTGRES: База данных:	ecss_storekeeper_db (значение по умолчанию)	
Введите домен для поиска:	test.domain (значение по умолчанию)	
POSTGRES: Количество запрашиваемых контактов:	10000 (значение по умолчанию)	
Хотите включить транслит для телефонной книги (LDAP)?	No (значение по умолчанию)	
Выберите пункты, которые хотите настроить:	**proxy-filter** (необходимо выбрать)	
Включить белый список для api/proxy?		

белый список адресов прописать в файл /etc/ecss/ecss-restfs/api-proxy.conf

```
# ------------------------------------------------------------
# Формат файла фильтра API (api/proxy):
#   - IP: 192.168.0.1
#   - Домен: example.com
#   - Wildcard: *.wildcard.com
#   - Комментарии: строки, начинающиеся с #
# Пример:
#   192.168.0.1
#   example.com
#   # комментарий
#   *.wildcard.com
# ------------------------------------------------------------
```

ВАЖНО!

В случае успешного проксирование (200) кеш хранится 28 дней, поэтому для применения конфигурации фильтра, в случае изменения, - кеш (/var/cache/ecss/restfs-api) стоит удалить и перезапустить сервис ecss-restfs.

после установки пакета ecss-restfs проверьте наличие wav файлов авто-информатора в директории /var/lib/ecss/restfs/system/sounds/ командой:

на обоих хостах

```
ls -la /var/lib/ecss/restfs/system/sounds/
```

наличие wav файлов в указной выше директории говорит о корректности установки пакета ecss-restfs.

Проверить доступность этих файлов извне, для этой проверки выполним команду:

```
wget http://ecss1:9990/system/sounds/ai_you.wav
```

пример

```
wget http://ecss1:9990/system/sounds/ai_you.wav
--2025-09-16 08:27:36--  http://ecss1:9990/system/sounds/ai_you.wav
Распознаётся ecss1 (ecss1)… 10.0.10.40
Подключение к ecss1 (ecss1)|10.0.10.40|:9990... соединение установлено.
HTTP-запрос отправлен. Ожидание ответа… 200 OK
Длина: 11670 (11K) [audio/x-wav]
Сохранение в: «ai_you.wav»

ai_you.wav.1                                  100%
[=============================================================================
===>]  11,40K  --.-KB/s    за 0s
```

```
2025-09-16 08:27:36 (81,5 MB/s) - «ai_you.wav» сохранён [11670/11670]
```

проверочные файлы можно удалить

```
rm ai_you.wav
```

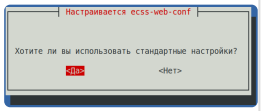
Установка media и web пакетов ecss и их настройка

```
sudo apt install -y ecss-media-server ecss-web-conf
```

Для медиасервера (*ecss-media-server/MSR*) возможно начальное конфигурирование с записью параметров в файл конфигурации,
для этого нужно провести конфигурацию **transport bind-addr,mcc bind-addres**:

Вопросы ecss-media-server	**Ответы для ecss1**	**Пример**
[MSR SIP] Введите bind-ip адрес (Enter)	**10.0.20.40** (необходимо ввести)	
[MSR Control-Channel] Введите bind-ip-адрес	**10.0.20.40** (необходимо ввести)	

Web-конфигуратор позволяет сделать управление системой более наглядным и комфортным. Установка web-конфигуратора не является обязательной, но рекомендуется.
Также при установке пакета **ecss-web-conf** автоматически устанавливается пакет **ecss-subsriber-portal-ui**. Приложение "Портал абонента" системы ECSS-10 позволяет абонентам системы самостоятельно управлять услугами, просматривать информацию по совершенным вызовам, активным конференциям, а также настраивать собственные IVR-скрипты для входящих вызовов. Описание работы веб-конфигуратора приведено в разделе "Портал абонента".

Вопросы ecss-web-conf	**Ответы для ecss1/ecss2**	**Пример**
Хотите ли вы использовать стандартные настройки?	Yes (значение по умолчанию)	

После формирования конфигураций по умолчанию переходим в директорию, в которой находятся файлы конфигурации и производим их проверку:

```
cd /etc/ecss/ecss-media-server/
```

```
cat /etc/ecss/ecss-media-server/config.xml
```

Конфигурация для msr: файл - config.xml, в директории conf.d файл default.xml.
По своей сути default.xml — это дополнение config.xml, которое определяет секцию аккаунтов. Это сделано для того, чтобы после обновлений пакета данная конфигурация оставалась неизменной. Вид config.xml представлен здесь: файл конфигурации.

```
<?xml version="1.0" encoding="utf-8"?>
<config date="02:30:33 07.12.2023">
  <general log-level="3" log-rotate="yes" max-calls="8192" max-in-group="512" load-sensor="medi
a" load-delta="10" calls-delta="100" spool-dir-size="100M" log-name="msr.log" log-path="/var/
log/ecss/media-server" use-srtp="disabled" enable-ice-transport="no" ice-update="no"
 aggressive-ice="yes" stun-server="" suspicious-mode="no"/>
  <transport bind-addr="10.0.20.40" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca" user="webrtc@live.com" password="muazkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60" rtcp-timeout="0" rtp-
timeout="350" udp-src-check="no" cn-multiplier="3" port-start="12000" port-range="2048" tias-
in-sdp="no" thread-cnt="2" vid-enc-threads="2" vid-dec-threads="2" video-conf-layout="evenly"
 keyframe-interval="1000" vid-decode-delay="100" silence-threshold="-30" dtmf-flash-disable="no
" video-dscp="0" other-dscp="0" dummy-video-src="/usr/share/ecss-media-server/video/
dummy_video.yuv" video-enc-width="640" video-enc-height="360" finalsilence="1000" rtcp-stat-
dump="yes" silent-codec-switch="yes"/>
  <codec pcma="1" pcmu="2" ilbc="0" gsm="0" g722="3" g726="0" g729="0" speex="0" l16="0" g7221="
0" opus="0" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
  <accounts>
    <!-- <dynamic msr_name="msr.name"
            realm="sip:127.0.0.1:5000"
            dtmf_mode="rfc+inband+info"
            auth_name="user"
            auth_password="password" /> -->
  </accounts>
  <pbyte>
    <mcc bind-addr="10.0.20.40" port="5700"/>
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

По умолчанию после инсталляции ECSS-10 **активизирует только** кодеки — **pcma, pcmu, g722,** h264, h263-1998, t38. Остальные имеют статус =0 = выкл. **Если требуется** активировать остальные кодеки, в редакторе nano измените приоритет с 0 на цифру приоритета (уровень приоритета не должен повторятся для сервиса audio/video/fax)
sudo nano /etc/ecss/ecss-media-server/config.xml

чтобы активизировать **все** кодеки
строчку -
<codec pcma="1" pcmu="2" ilbc="**0**" gsm="**0**" g722="3" g726="**0**" g729="**0**" speex="**0**" l16="**0**" g7221="**0**" opus="**0**" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
привести к виду -
<codec pcma="1" pcmu="2" ilbc="**6**" gsm="**4**" g722="3" g726="**11**" g729="**5**" speex="**7**" l16="**8**" g7221="**9**" opus="**10**" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>

Недопустимо указывать у разных кодеков одинаковый приоритет, для примера g722="5" g729="5". В этом случае media-server не сможет работать корректно.
За исключением "0" - выключено.

Рассмотрим секцию аккаунтов (файл default.xml):

```
cat /etc/ecss/ecss-media-server/conf.d/default.xml
```

После инсталляции пакетов в фале будет указан ip-add 127.0.0.1.

```
<?xml version="1.0"?>
<config>
      <accounts>
            <dynamic msr_name="msr.ecss1" realm="sip:127.0.0.1:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>
      </accounts>
</config>
```

Требуется указать в конфигурации, реальный IP-add.

Необходимо привести содержимое файла к виду указанному снизу (в примере ecss1 имеет IP-add 10.0.20.40):

на обоих хостах

```
sudo nano /etc/ecss/ecss-media-server/conf.d/default.xml
```

Настройка msr для ecss1 (/etc/ecss/ecss-media-server/conf.d/default.xml)

Проверить на корректность данных.

```
<?xml version="1.0"?>
<config>
  <accounts>
    <dynamic msr_name="msr.ecss1" realm="sip:10.0.20.40:5000" dtmf_mode="rfc+inband+info"
 auth_name="user" auth_password="password">
      <via dynamic-ifaces="no">
        <iface name="net.20" ip="10.0.20.40"/>
      </via>
    </dynamic>
  </accounts>
</config>
```

Указываем текущие настройки, согласно которым происходит регистрация msr на core.

Основными параметрами здесь являются: msr_name и realm:

- msr_name — параметр, определяющий название msr. (рекомендуется задавать msr. и к какому хосту он принадлежит, например msr.ecss1);
- realm — определяет адрес для регистрации на ядре. Точка входа по умолчанию — порт 5000.

После изменения конфигурации файлов **/etc/ecss/ecss-media-server/config.xml** и **/etc/ecss/ecss-media-server/conf.d/default.xml** необходимо выполнить перезапуск сервиса **ecss-media-server**, чтобы изменения вступили в силу, следующей командой:

```
sudo systemctl restart ecss-media-server
```

Дальнейшая настройка

snpmd

Настройка SNMP описана в разделе "Управление и мониторинг по протоколу SNMP".

Настройка безопасности. SSH

Выполните настройку сервера SSH:

```
sudo nano /etc/ssh/sshd_config
```

Укажите в конфигурационном файле порт и адрес, по которому можно обратиться к серверу:

Настройка ssh для ecss1 (/etc/ssh/sshd_config)

```
# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Port 2000
#AddressFamily any
ListenAddress 10.0.10.40
#ListenAddress ::

<...>
```

Перезапустите ssh:

```
sudo systemctl restart ssh.service
```

Перезагрузка хоста

```
sudo reboot
```

Добавление услуг

Установка всех услуг указных в лицензии:

выполняется в CoCon

```
/cluster/storage/ds1/ss/install ds1@ecss1 ss_*
```

Результат:

```
/cluster/storage/ds1/ss/install ds1@ecss1 ss_*
Successfully installed: /var/lib/ecss/ss/ss_dnd.xml
Successfully installed: /var/lib/ecss/ss/ss_clir.xml
...
Successfully installed: /var/lib/ecss/ss/ss_chunt.xml
```

13.6 Первоначальная конфигурация системы с помощью web-конфигуратора

пример первоначальной конфигурации через Web интерефейс

13.6.1 Исходные данные

- Установка системы завершена;
- Система готова к дальнейшей настройке;
- Интерфейсы запущены.

Рекомендуется использовать последние доступные версии браузеров. Рекомендуемые к использованию браузеры: Яндекс, Chrome.

Для начала конфигурации системы необходимо зайти в web-конфигуратор по адресу http://<Ipadd SSW>.

Для определения и регистрации в системе планируются:

- Абоненты с номерами 42001, 42002 , 42003, 42004 ,42005, 42006, 42007, 42008, 42009, 42010;
- Транк в сторону шлюза.

13.6.2 Подготовка к работе

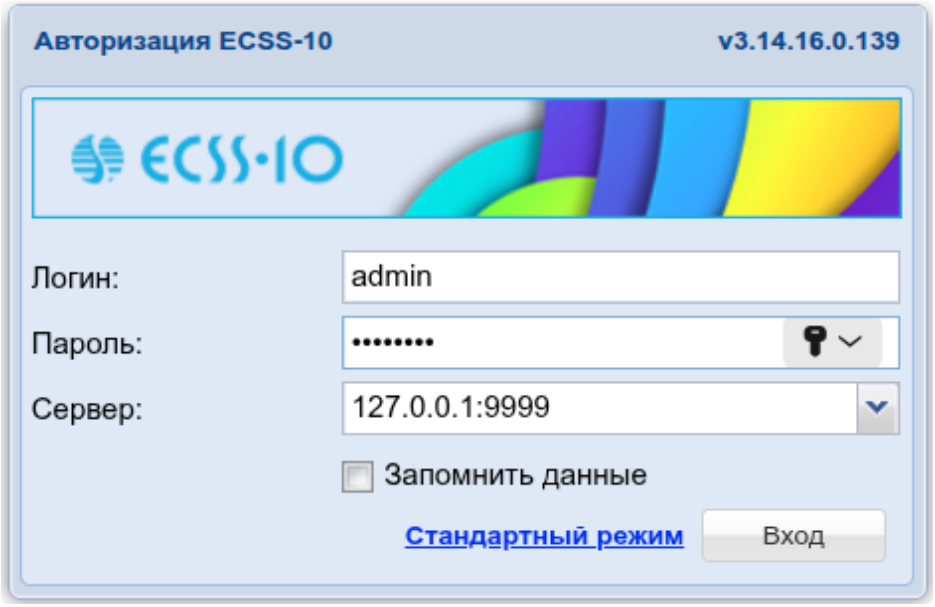


Рисунок 3 — Вход в web-конфигуратор (окно авторизации)

В окне авторизации необходимо ввести значения, определенные при установке web-конфигуратора.

⚠ Стандартные значения для авторизации:
Логин: admin
Пароль: password

После входа в систему будет видна основная рабочая область с иконками приложений, а также строка состояния с доступными опциями, в частности:

- 1 — выход из системы;
- 2 — выбор домена;
- 3 — выбор языка.

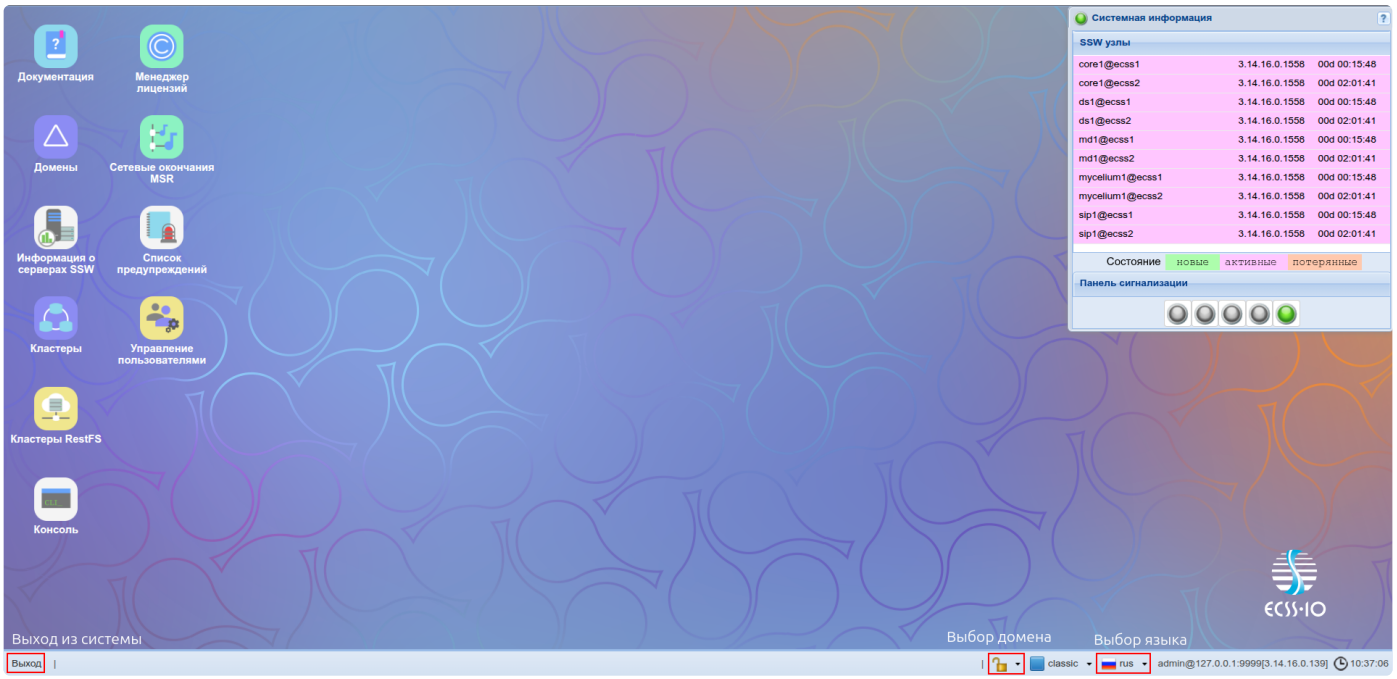


Рисунок 4 — Вид рабочей области web-конфигуратора

13.6.3 Создание учетной записи оператора

После авторизации для повышения безопасности при эксплуатации программного коммутатора рекомендуется создание учетных записей для операторов, а также сменить пароль для пользователя **admin**.

Для создания новой учетной записи оператора используется приложение "Управление пользователями" ("User manager"):

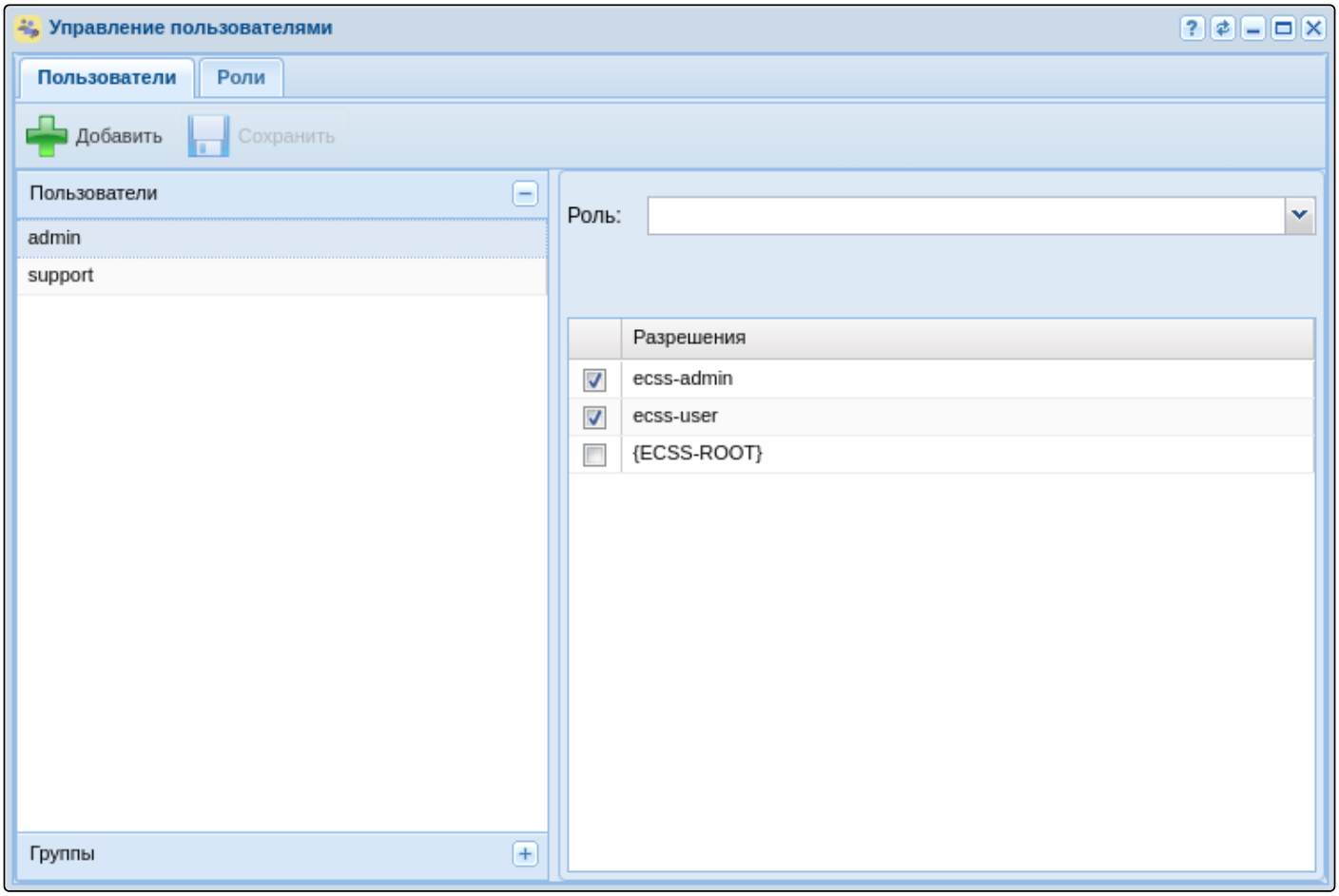


Рисунок 5 – Вид приложения "*Управление пользователями*"

Нажмите кнопку "*Добавить*"  . В открывшемся окне требуется определить новую учетную запись, для этого:

1. В поле "*Имя*" введите логин учетной записи, например "test";
2. В поле "*Пароль*" и "*Подтверждение*" введите пароль для пользователя, например "testpassword";
3. Определите уровень прав доступа для пользователя через выбор текущих разрешений или с помощью ролей, например **ecss-user**. Описание можно посмотреть в разделе "Управление пользователями".

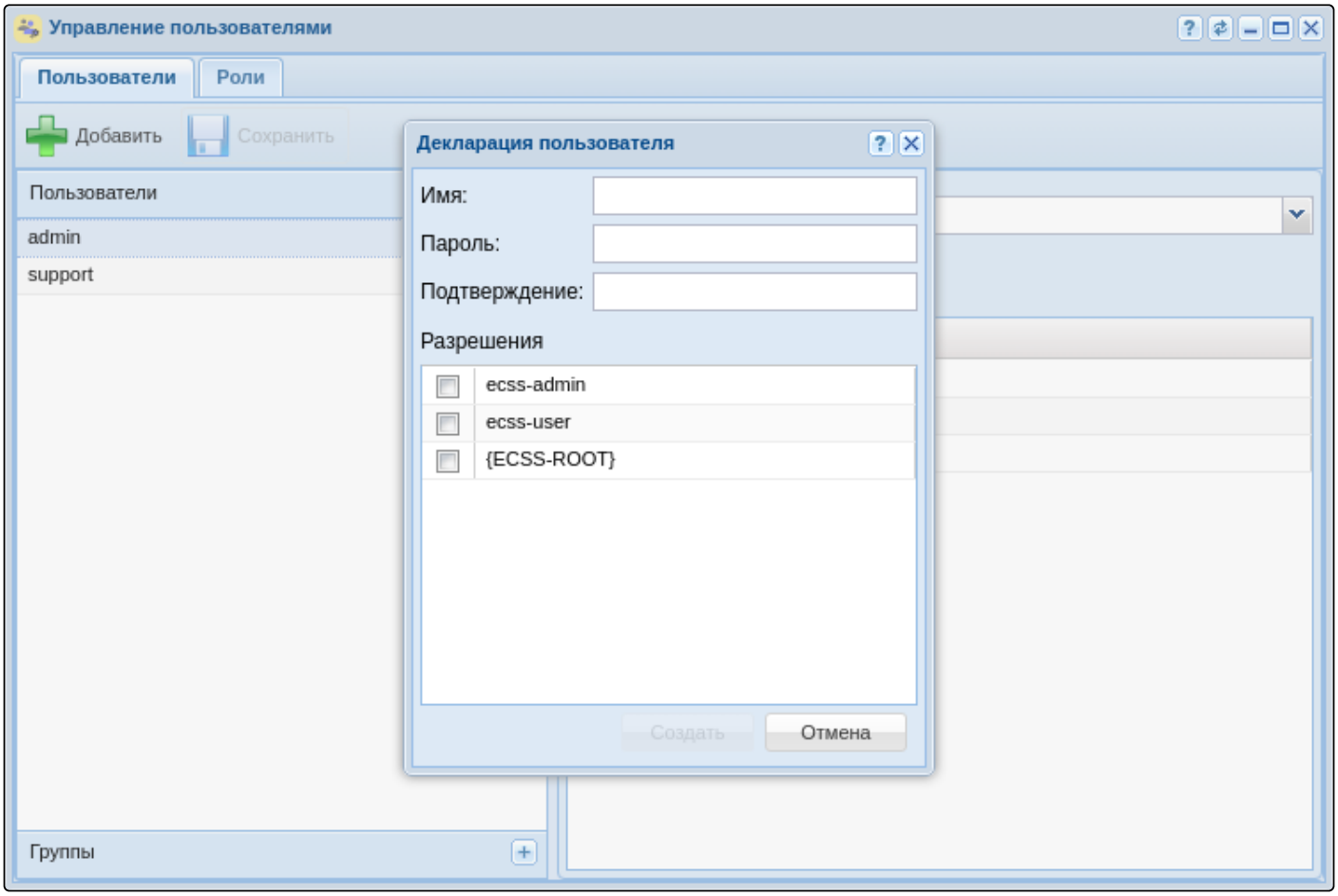


Рисунок 6 – Диалоговое окно создания учетной записи оператора

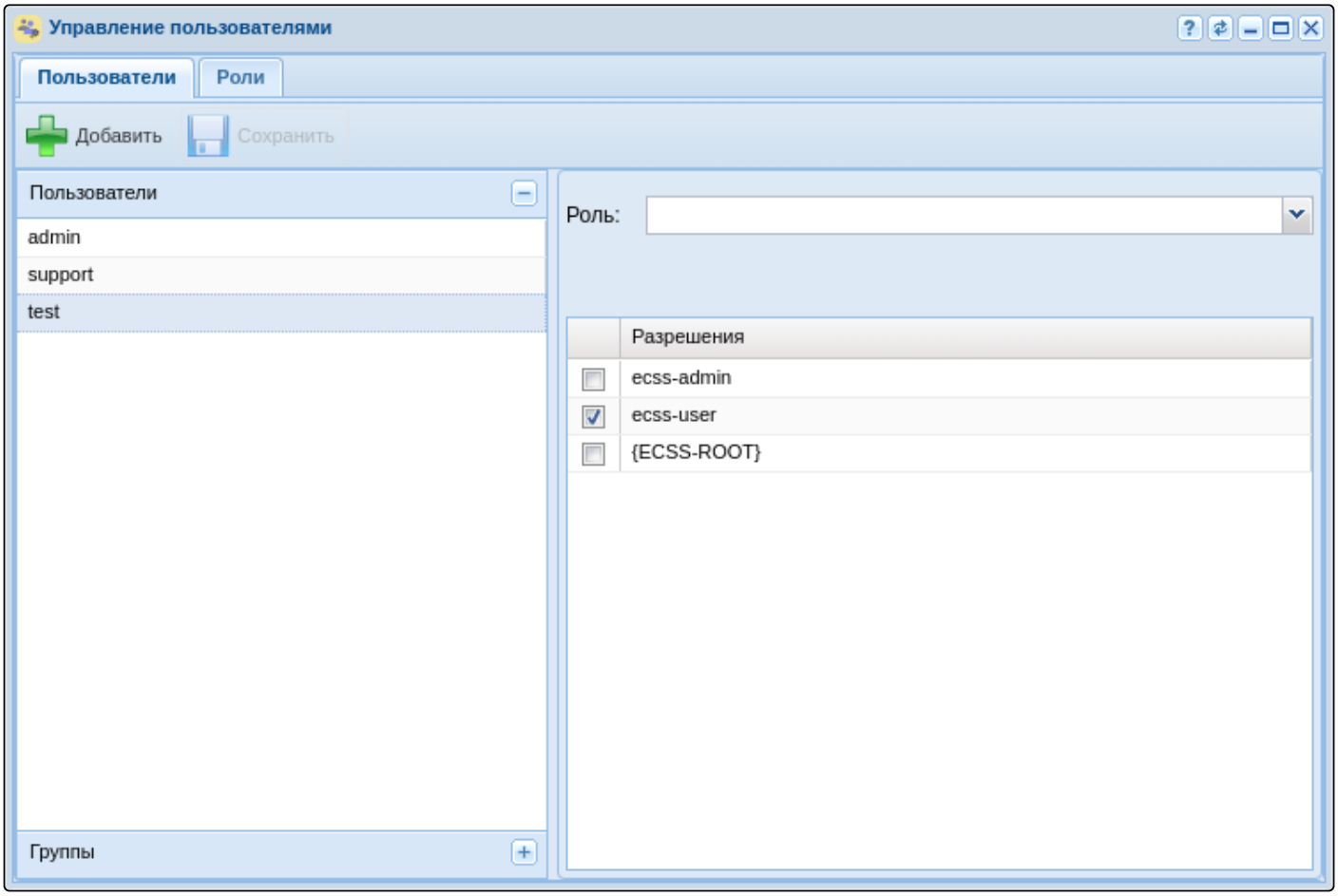


Рисунок 7 — Вид приложения с созданной учетной записью оператора

Для смены пароля рядом с названием пользователя нажмите на кнопку редактировать. В появившемся диалоговом окне введите:

1. Старый пароль (для пользователя "**admin**" по умолчанию паролем является "**password**");
2. Новый пароль;
3. Подтвердите новый пароль.

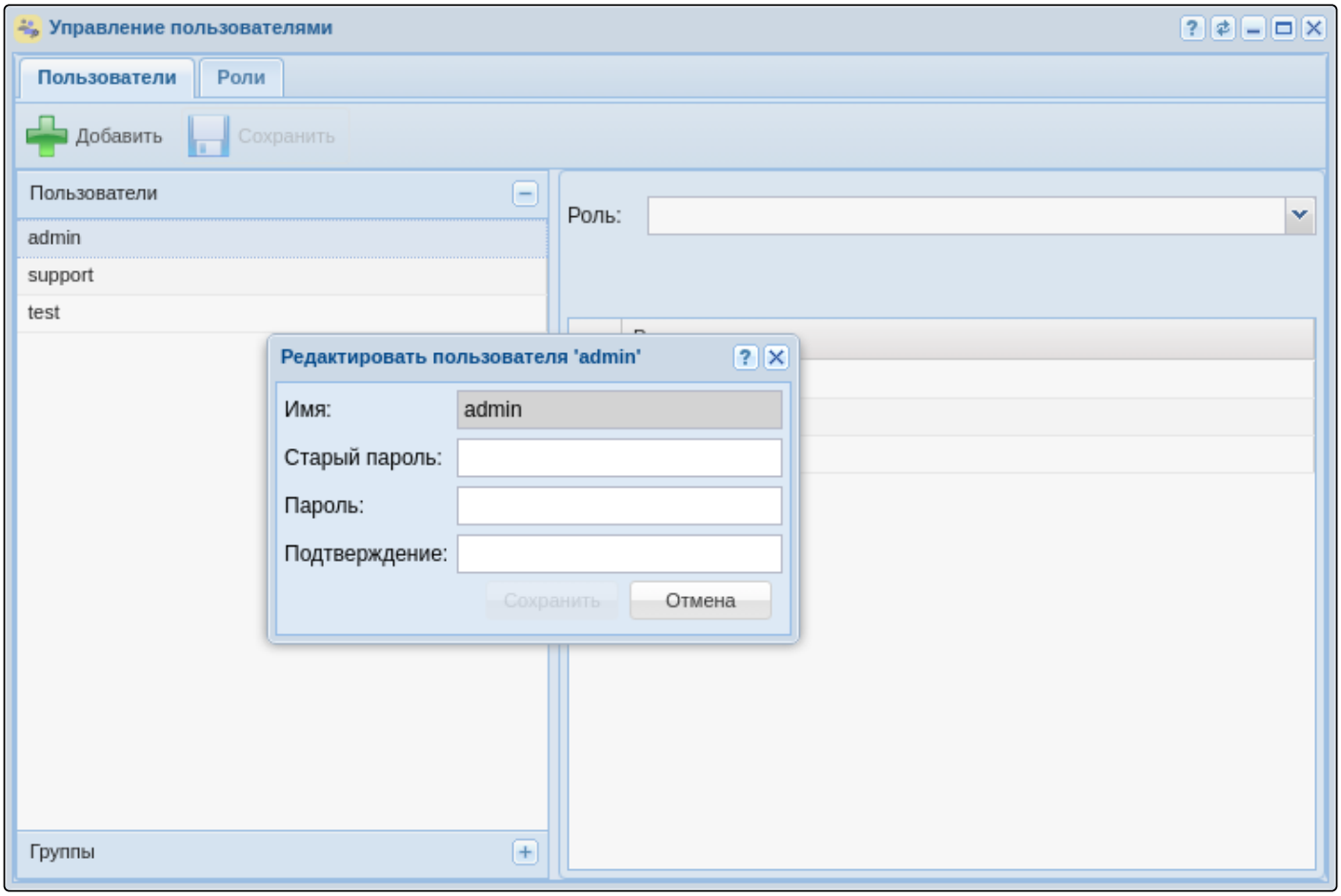
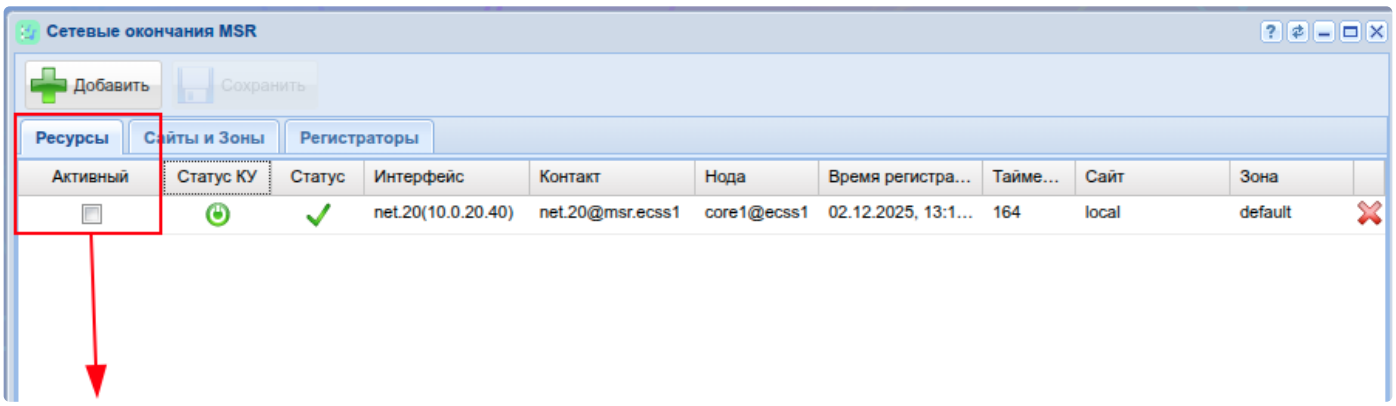


Рисунок 8 — Диалоговое окно редактирования учетной записи

13.6.4 Добавление медиа-ресурсов.

Для того чтобы начать работу с медиа-сервером сразу после его регистрации, необходимо активировать сетевые интерфейсы MSR. Приложение web-конфигуратора "Сетевые окончания MSR" ("MSR registrars") служит для настройки медиа-ресурсов.



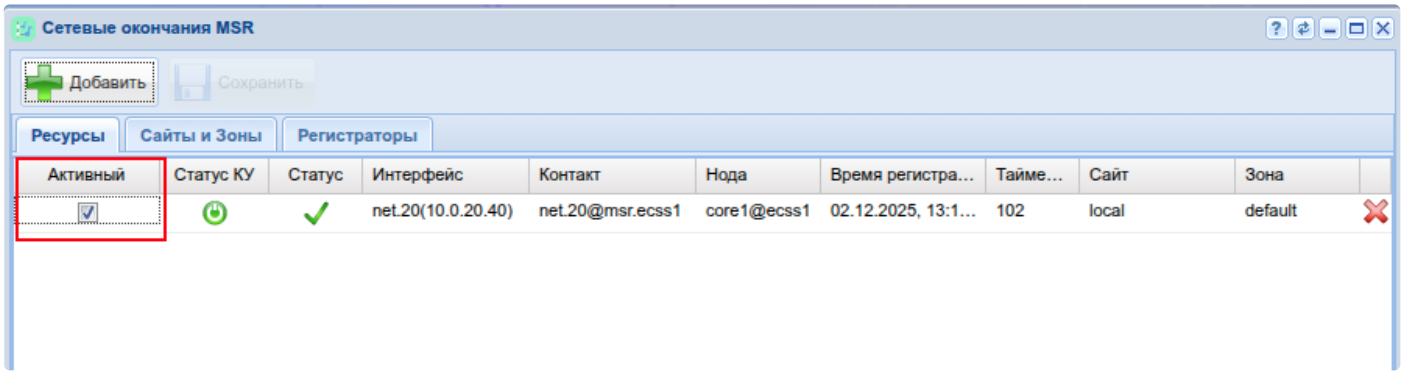
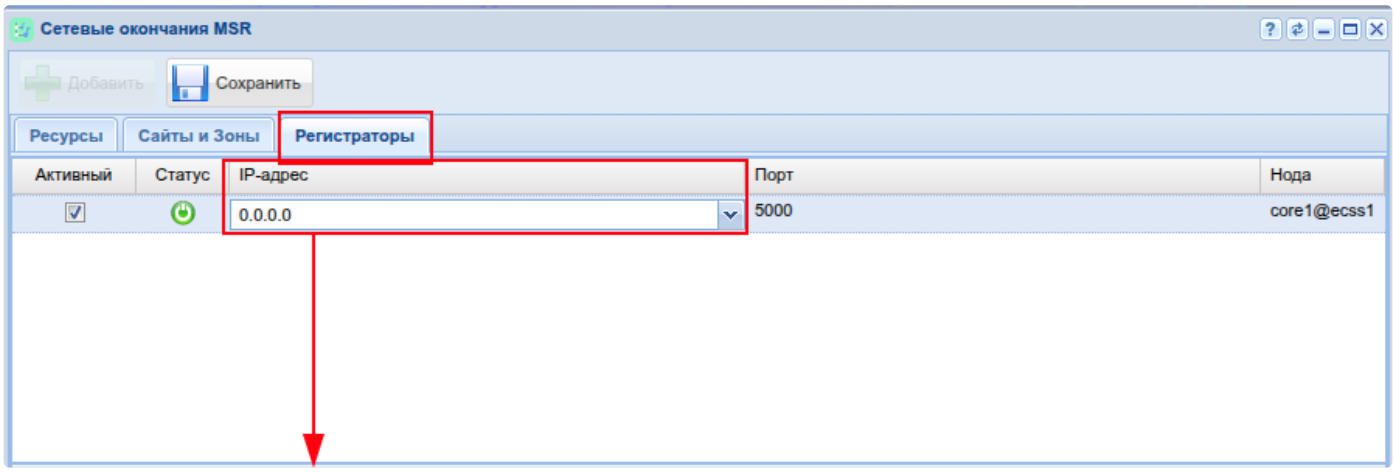
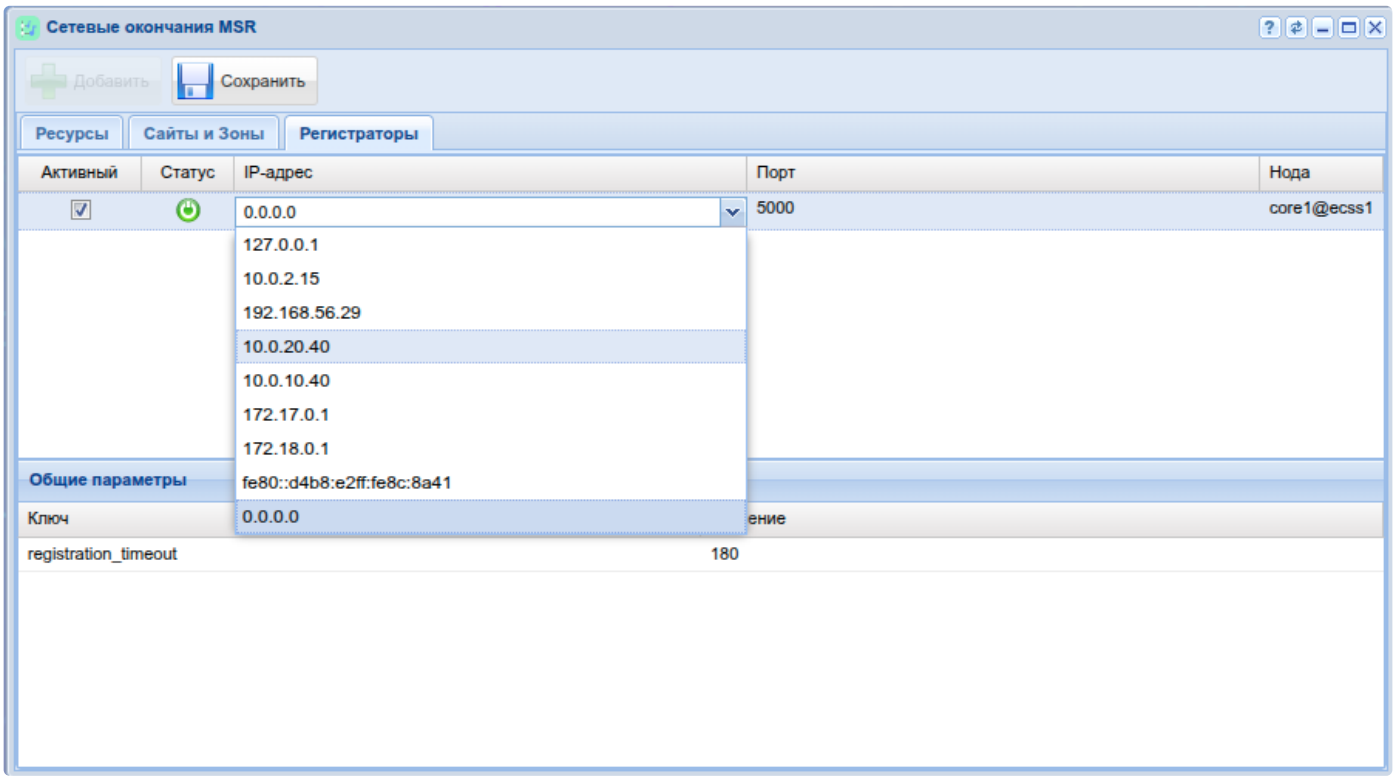


Рисунок 9 – Сетевые окончания MSR

Установить ipадрес для приема соединений от медиасервера, изменив значение 0.0.0.0 на реальные адреса MSR:





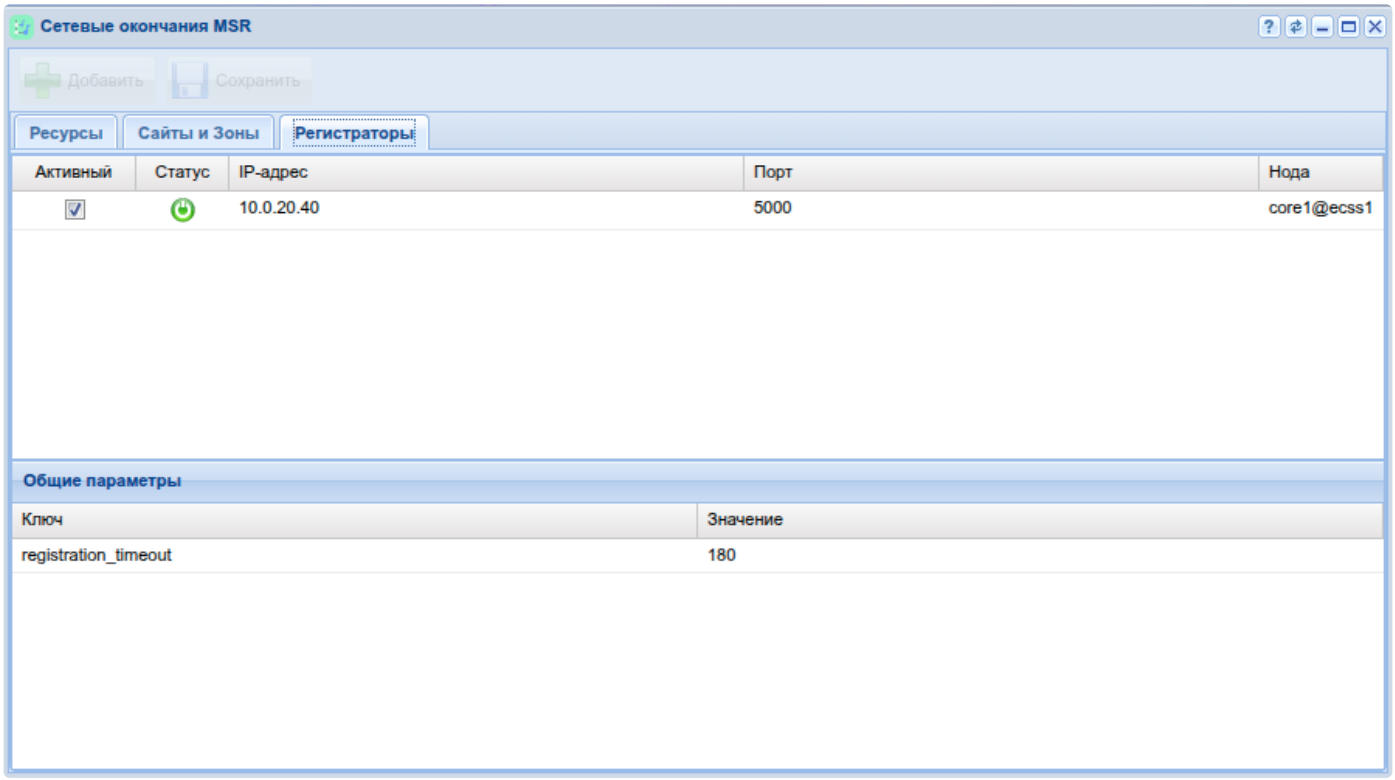


Рисунок 10 — Установка IP-add регистратора MSR

13.6.5 Создание IP-set (sip-транспорта) и присвоение его к домену

IP-set — совокупность одного или нескольких IP-адресов из одной подсети и списка слушающих UDP/TCP-портов, которые будут открыты SIP-адаптером на данных адресах.

Для настройки интерфейса необходимо открыть приложение "Кластеры ("Clusters").

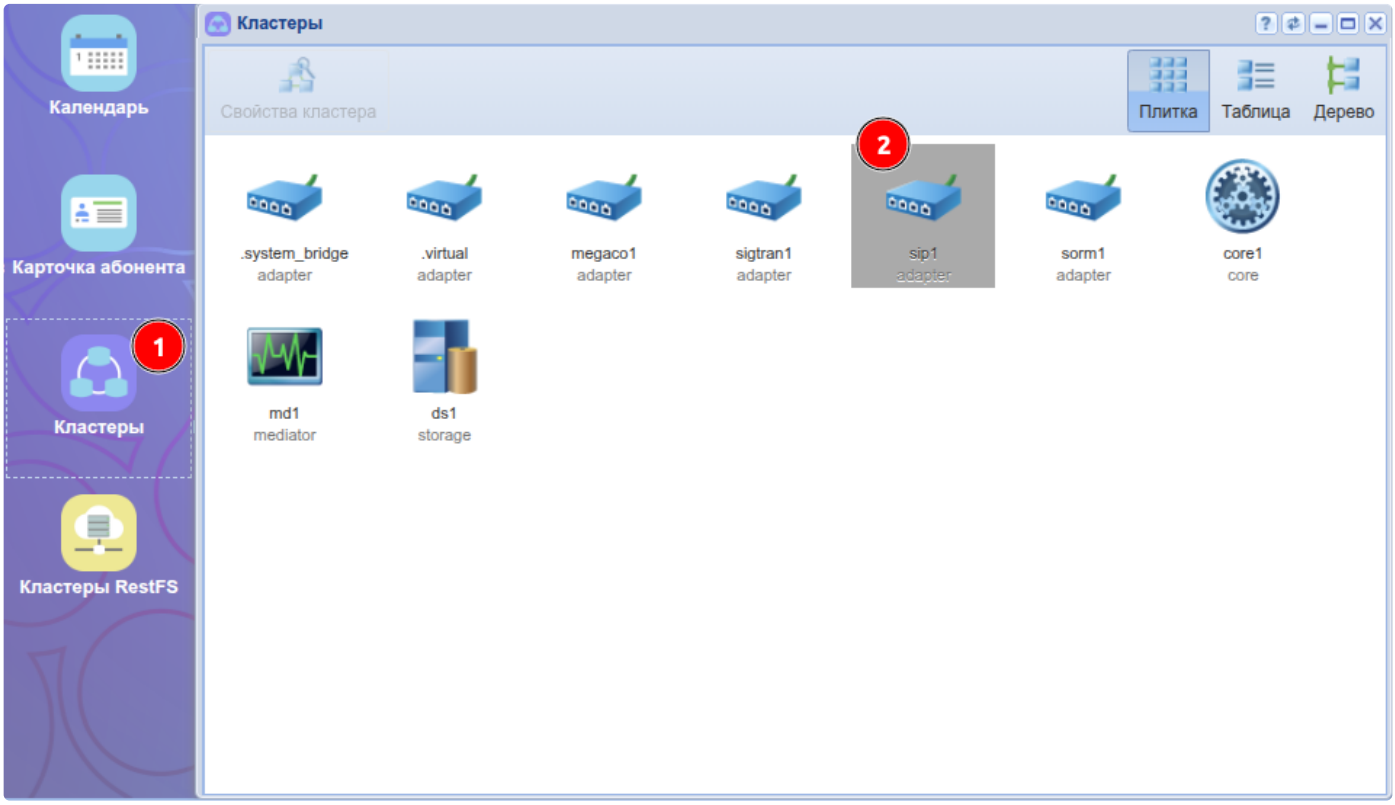


Рисунок 11 — Вид приложения "Кластеры"

Для создания новой группы IP-адресов (IP-set) выделите кластер SIP-адаптера "sip1" и нажмите на кнопку *"Свойства кластера"* (или нажмите на иконку кластера двойным щелчком левой кнопки мыши).

В появившемся диалоговом окне перейдите на вкладку *"Транспорт"*. Далее нажмите кнопку *"Добавить"*. Появится новая группа, для редактирования полей двойным щелчком кнопки мыши нажмите на нужное:

1. Переименуйте группу адресов (IP-set), например "test_set";

2. Задайте порт, по которому будет происходить обращение к домену, например 5060;

3. Разверните вновь созданную группу нажатием на треугольник слева от названия группы;

4. Определите адрес для ноды SIP-адаптера, согласно примеру конфигурации. Для примера 10.0.20.40:

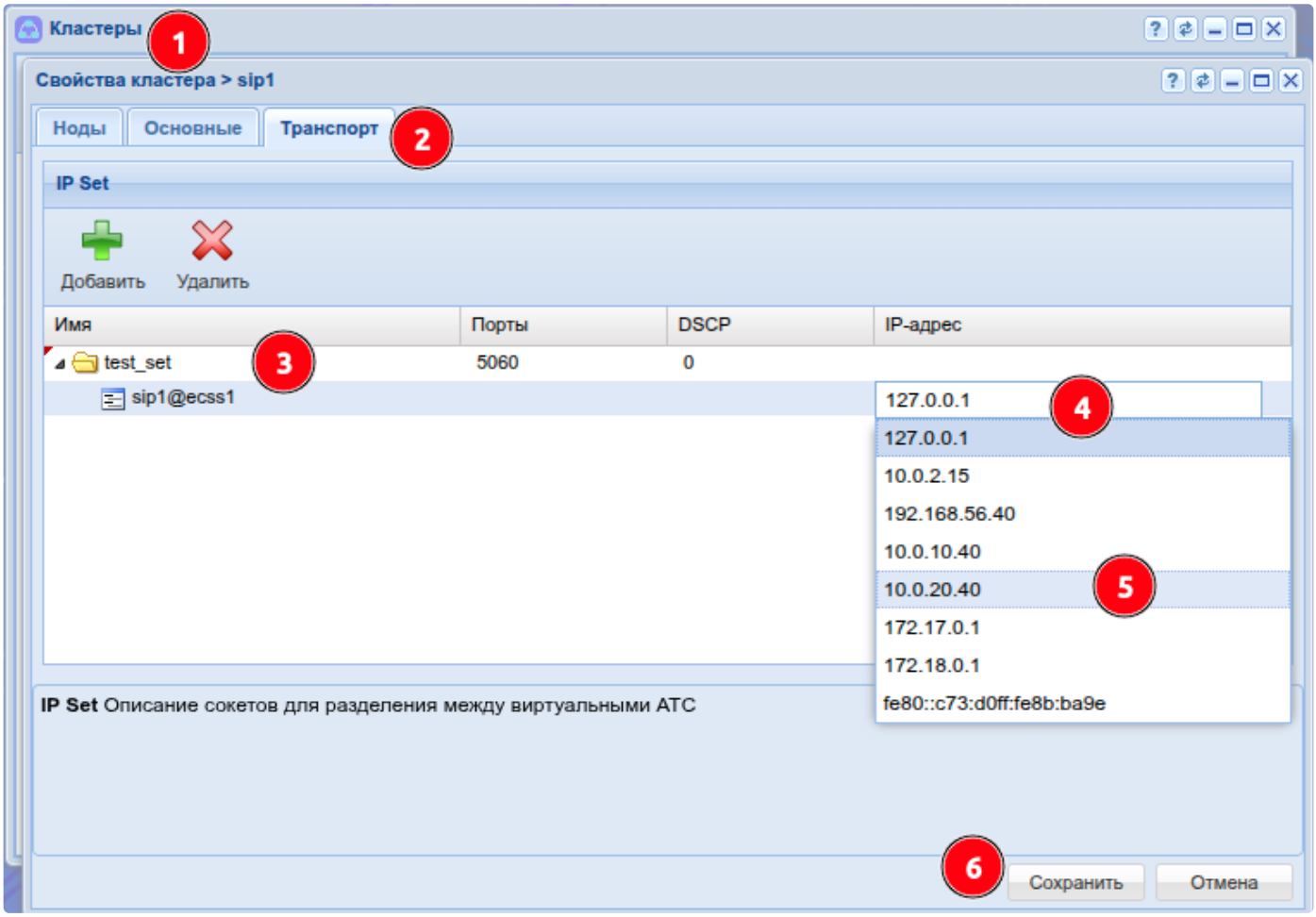


Рисунок 12 — Вид приложения "Кластеры/Транспорт"

Нажмите кнопку *"Сохранить"* для применения настроек.

13.6.6 Создание домена

Домен - Виртуальная PBX внутри нашего SSW. Таких виртуальных PBX может быть создано необходимое для проекта кол-во.

Для создания домена необходимо войти в приложение *"Домены" ("Domain")*. В открывшемся окне требуется создать домен, для этого:

1. Нажмите кнопку "*Добавить домен*":

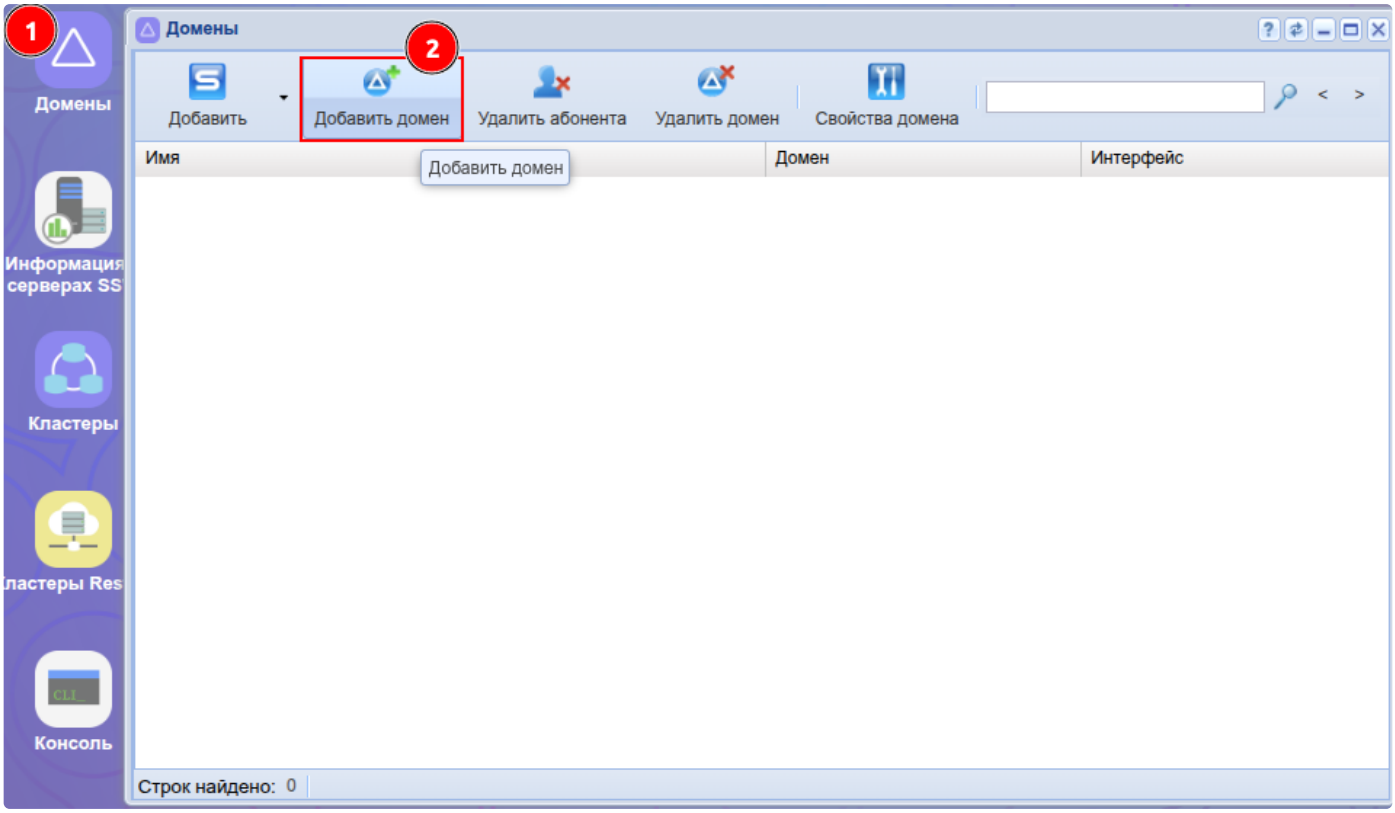


Рисунок 13 — Добавление домена в систему

2. В появившемся диалоговом окне доступны следующие настройки:

- *Имя (Name)* — индивидуальное имя виртуальной АТС;
- *Профиль услуг (SS profile)* — системный профиль дополнительных услуг. Этот профиль будет скопирован с тем же именем во вновь создаваемый домен и для всех услуг из этого профиля будет автоматически разрешён доступ через access-list;
- *IVR профиль (IVR profile)* — профиль IVR, заданный в приложении "Редактор IVR ограничений".

Введите название домена, например "test_domain";

3. Нажмите кнопку "*Ок*":

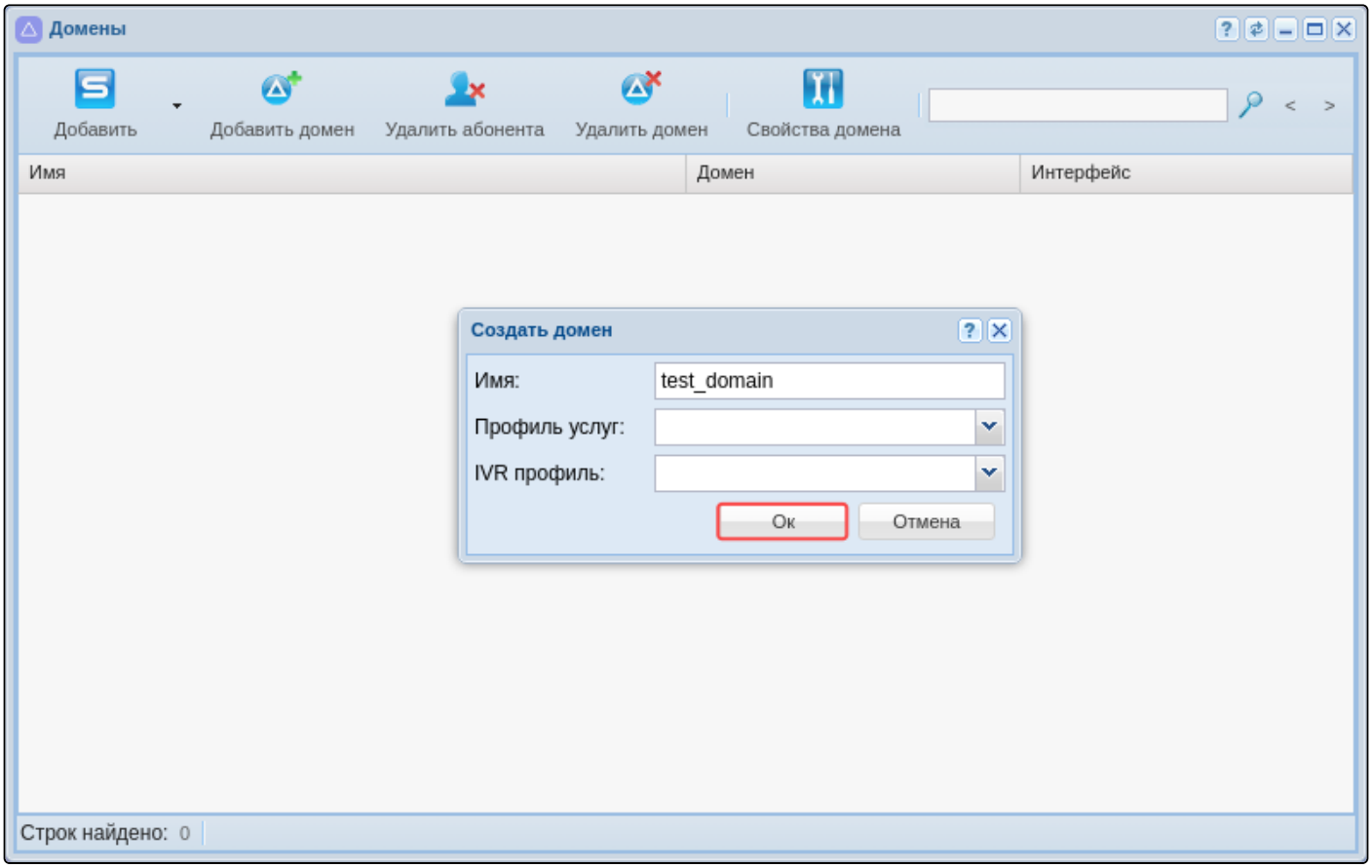


Рисунок 14 – Определение настроек нового домена

4. Нажмите кнопку *"Обновить"* в приложении.

Созданный домен будет отображаться в текущей конфигурации:

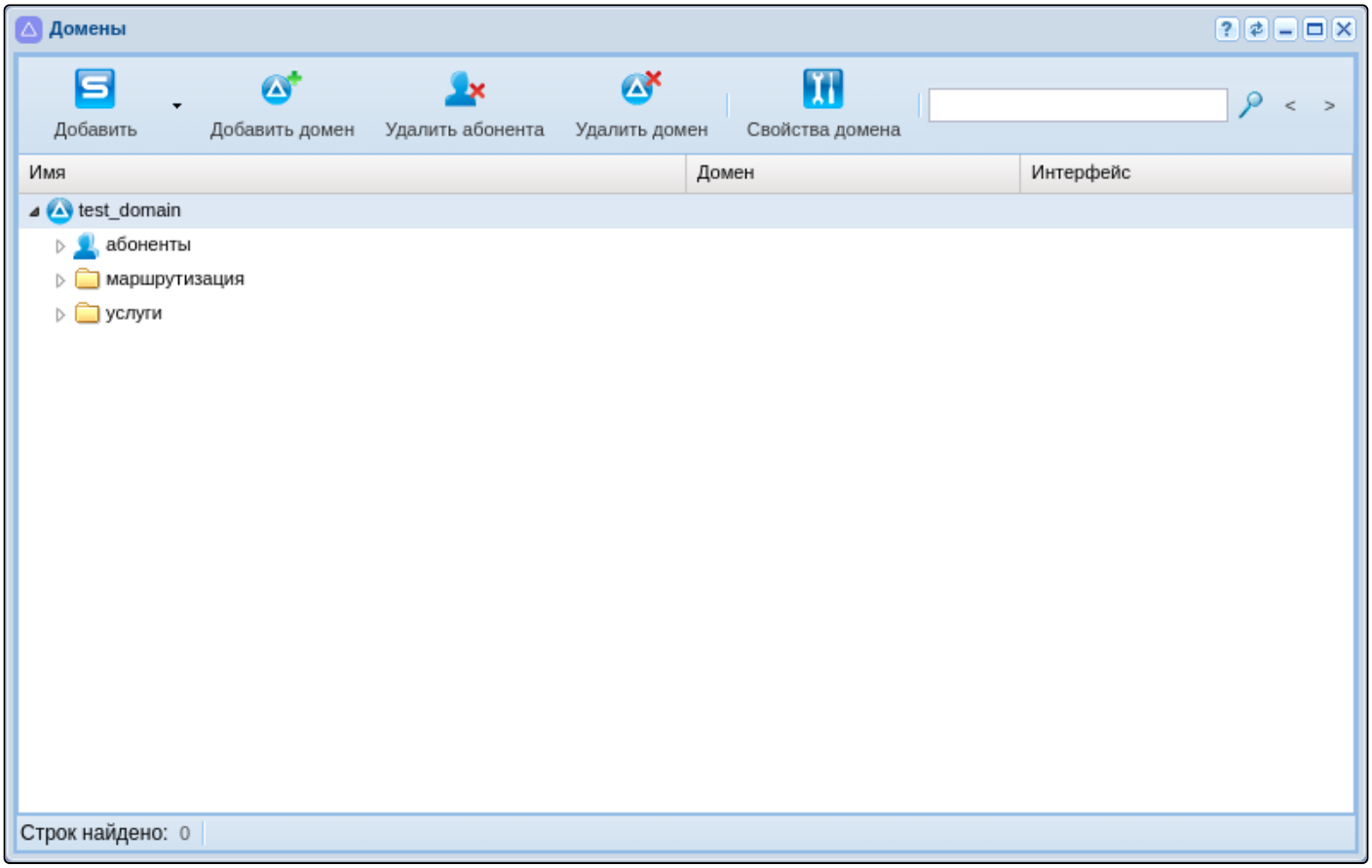


Рисунок 15 — Отображение созданного домена

Для редактирования текущего домена его необходимо выбрать в системе. Для перехода в домен воспользуйтесь опцией выбора домена (см. пункт 2 на рисунке "Вид рабочей области web-конфигуратора").

После выбора домена, согласно текущей конфигурации системы, станут доступны все приложения:



Рисунок 16 — Отображение приложений в текущей конфигурации системы

Для того чтобы **привязать группу адресов к домену**, необходимо вернуться в приложение *"Домены" ("Domain")*, выделить домен и перейти в настройки нажатием на кнопку *"Свойства домена"* или двойным нажатием левой кнопки мыши на домен.

В настройках списка откройте ветку "SIP", в ней "SIP транспорт", далее выберите в поле IP set созданную группу адресов. Нажмите кнопку *"Сохранить"* для применения настроек.

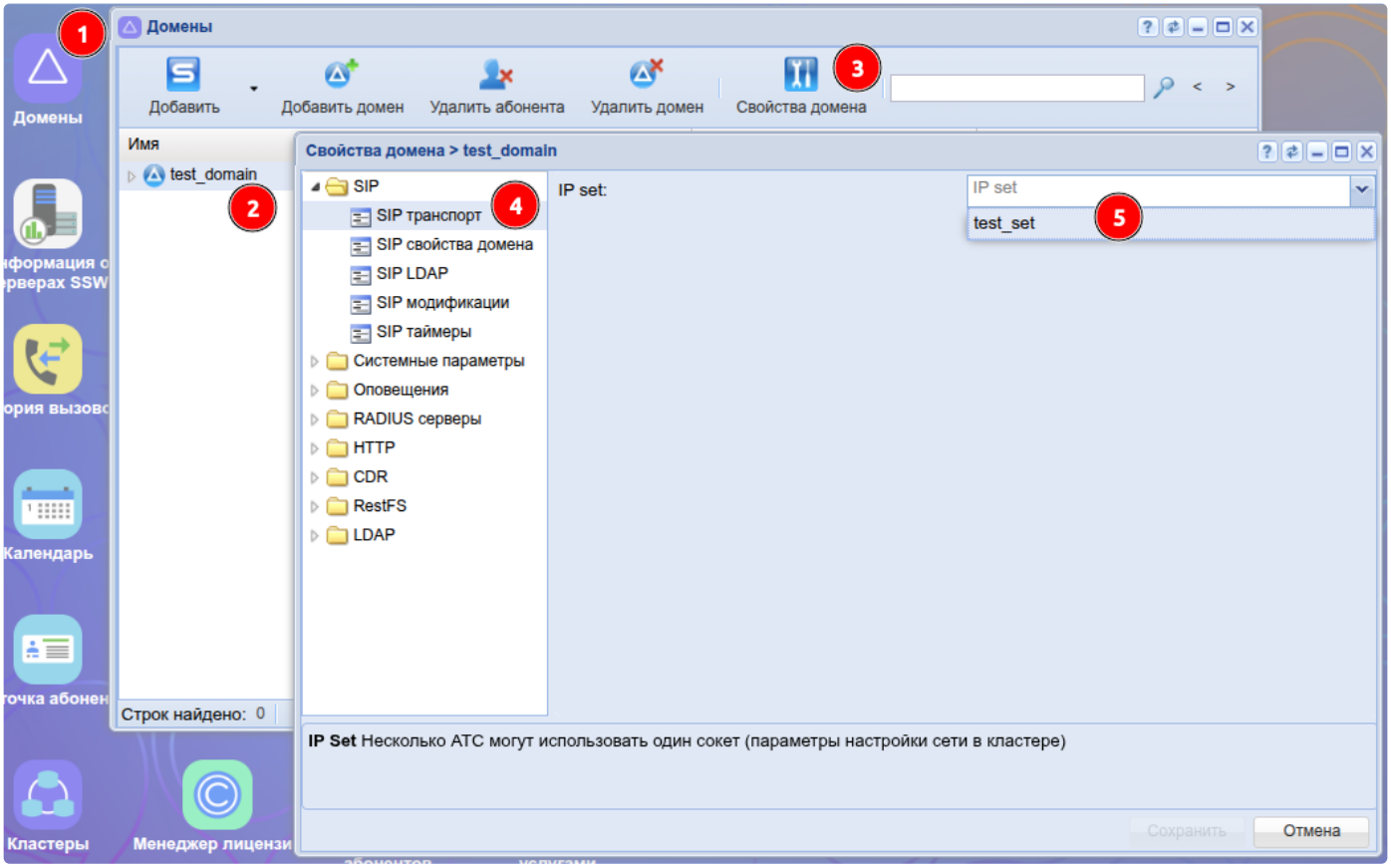


Рисунок 17 — Окно настройки для SIP транспорта

Выбрать требуемые услуги для созданного домена, которые будут использоваться (для примера для абонентского домена транковые услуги не востребованы , и наоборот для транзитного домена не нужно указывать услуги для абонента).

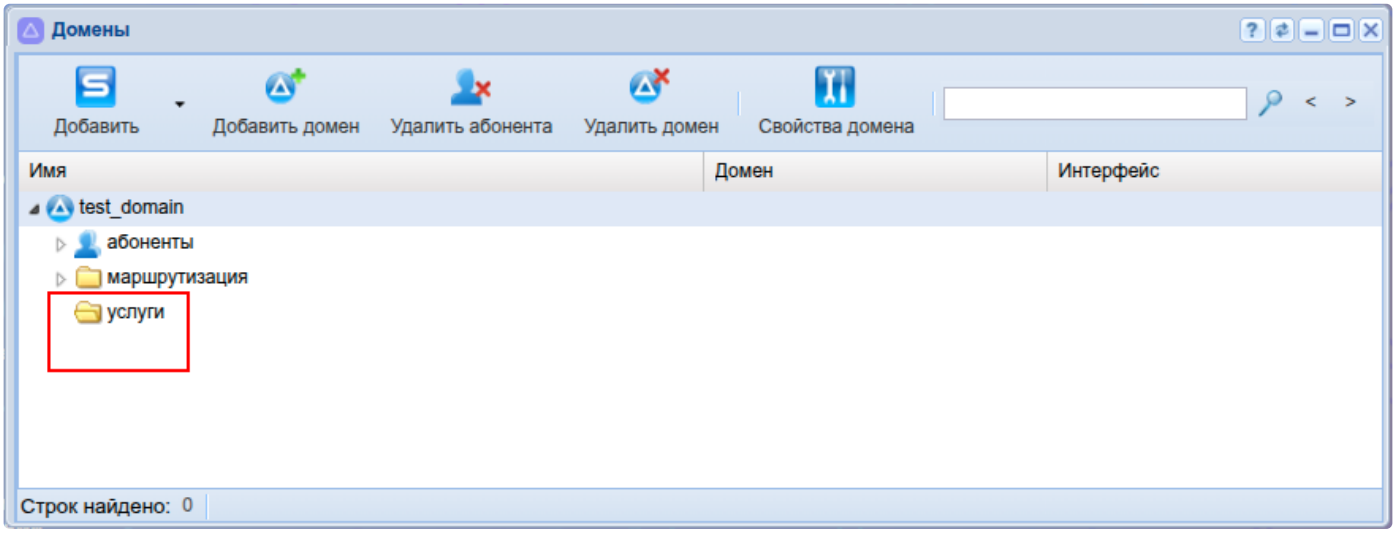


Рисунок 18 — После создания нового домена, услуги отсутствуют.

1. Выбрать приложение "Управление услугами"
2. Выбрать колонку "Установленные услуги" (услуги указанные в лицензии для SSW)

3. Выбрать необходимые для указанного домена услуги
4. Сохранить конфигурацию

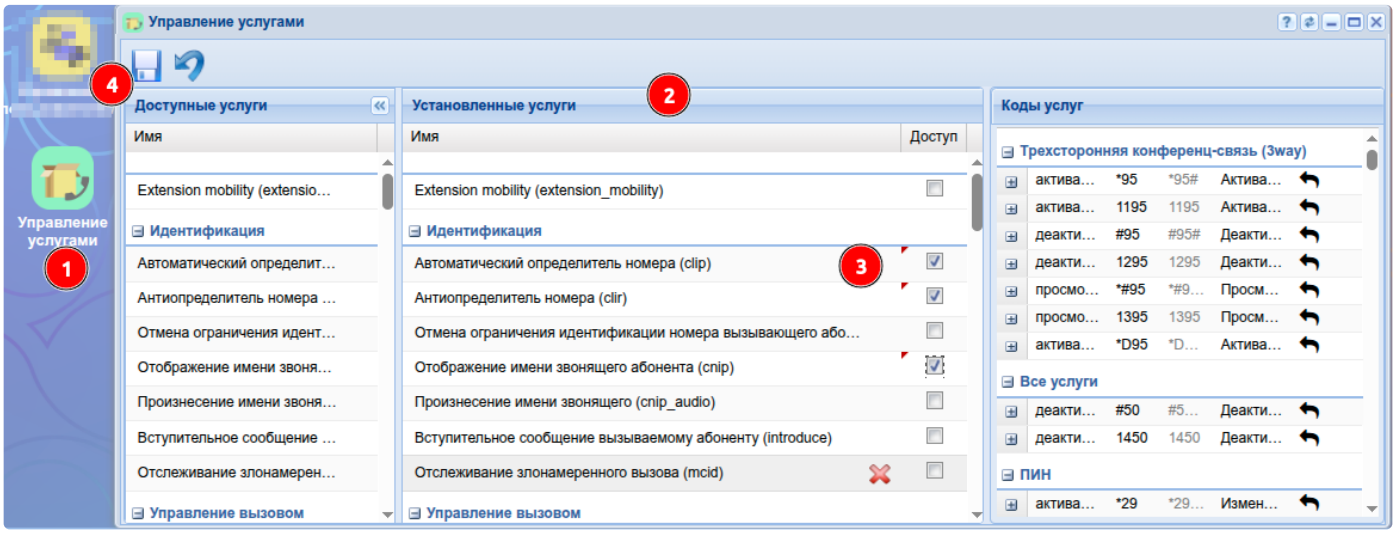


Рисунок 19 — Выбор услуг для абонентов / транков нового домена.

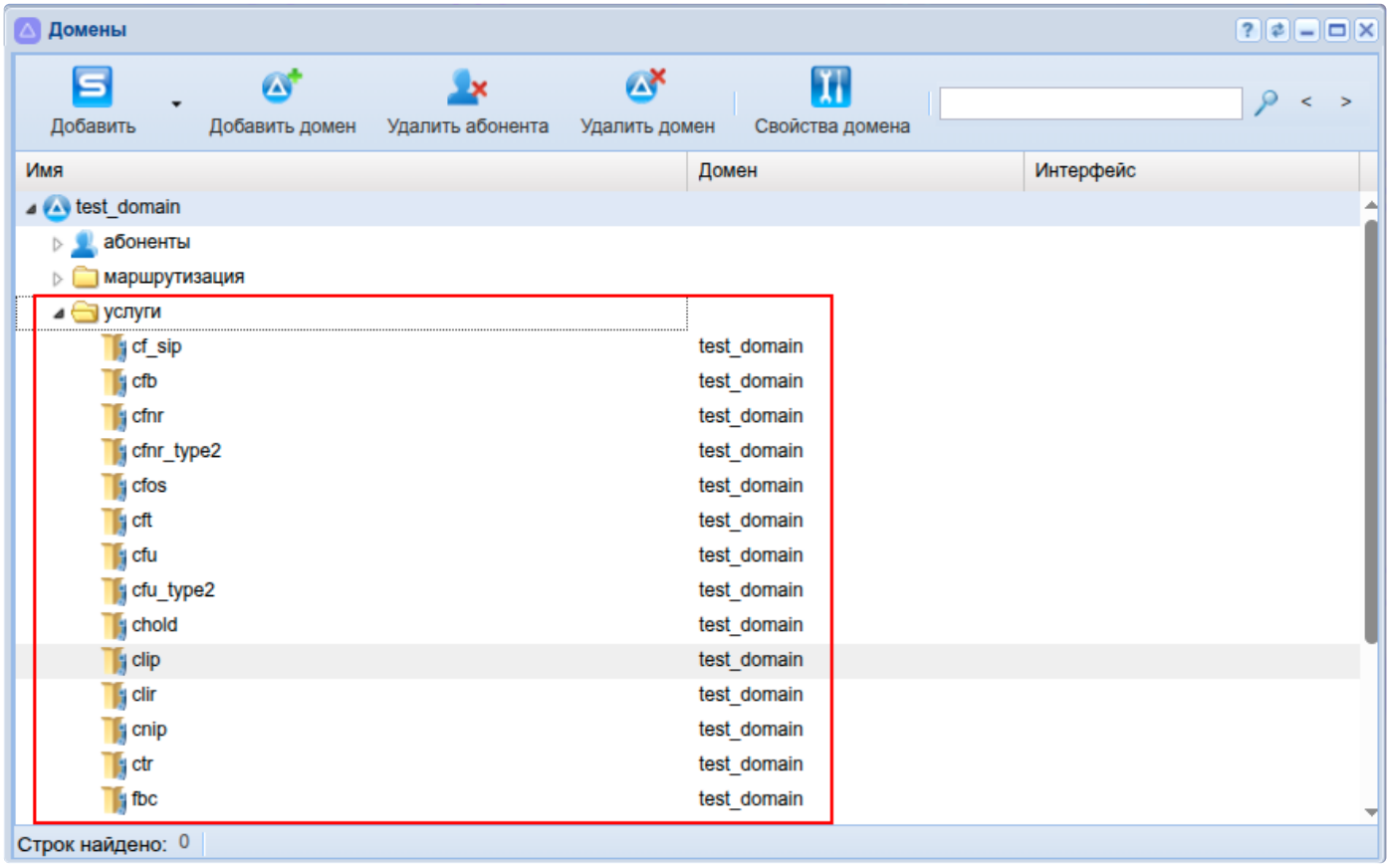


Рисунок 20 — Список услуг теперь доступных для абонентов нового домена.

13.6.7 Создание абонентов

Для создания и редактирования параметров абонентов в системе используется приложение "Карточка абонента" ("Subscriber card").

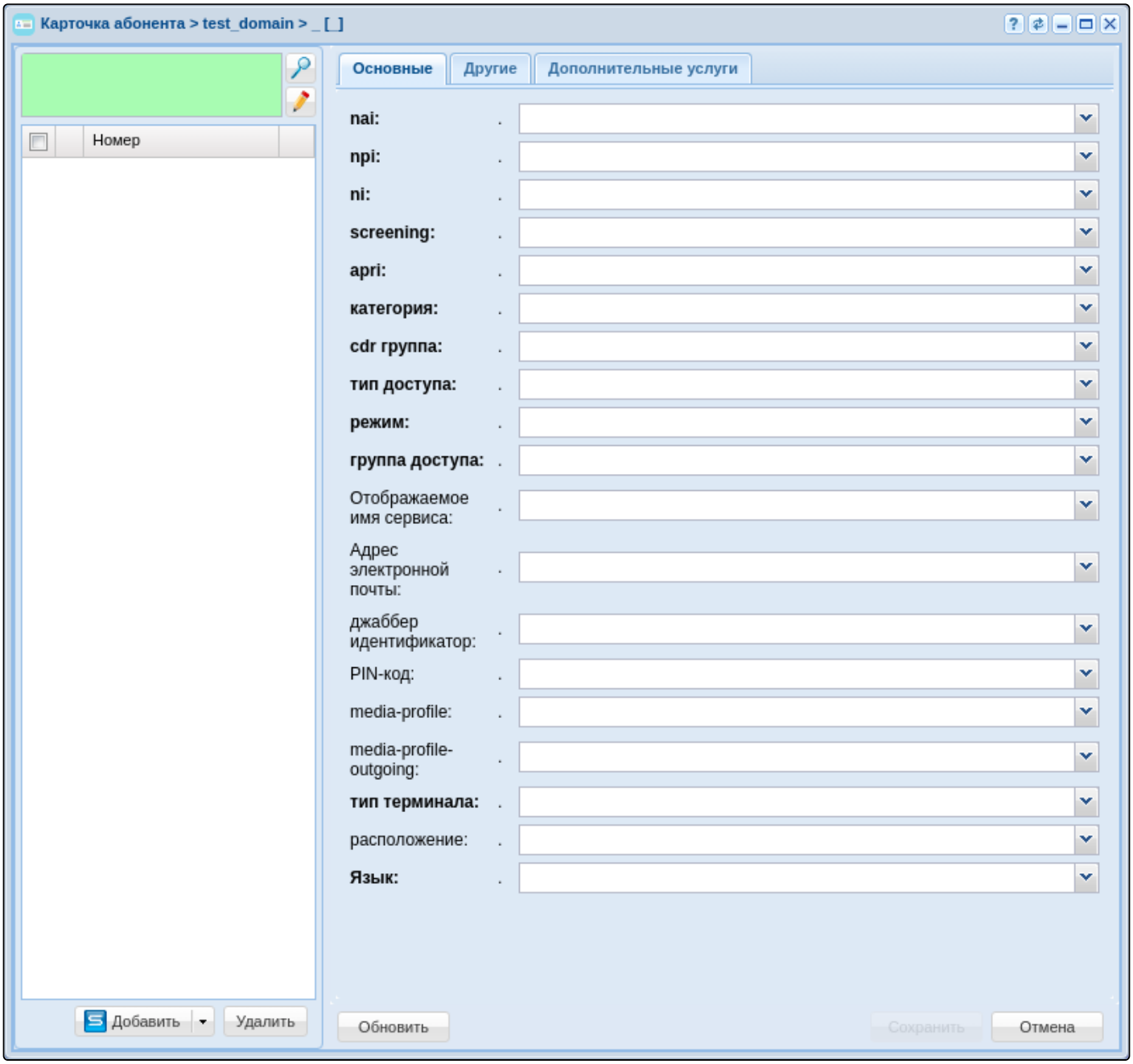


Рисунок 21 — Вид приложения "Карточка абонента"

В конфигурации возможно создание SIP-абонентов и виртуальных абонентов.

Для пользователей, имеющих физическое окончание, используется функционал SIP-абонента, в то время как виртуальный абонент используется тогда, когда нужна функциональность без физических окончаний. Например, номер для обращения к ivr-скрипту.

Для создания новых абонентов нужно нажать на кнопку *"Добавить"*.

В появившемся диалоговом окне укажите следующие параметры:

- *Имя интерфейса* — номер или группа номеров, который присваивается абоненту, строка, например {42001-42010} или abcd;
- *Контекст* — контекст маршрутизации, выбираем тот, который создали, например "test_name", или по умолчанию - "default_routing";
- Группа интерфейсов — имя группы объединяющей нескольких абонентов, для примера "sip";
- *Алиас как пользователь* — настройка, которая связывает номер сущности, алиас и пользователя одним и тем же названием, в примере настройка активирована;

- *Авторизация* — порядок проверки подлинности прав пользователя на доступ к данным, в примере используется"register";
- *Логин* — использую настройку номер как логин для простоты;
- *Пароль* — можно задать или использовать сгенерированный системой.

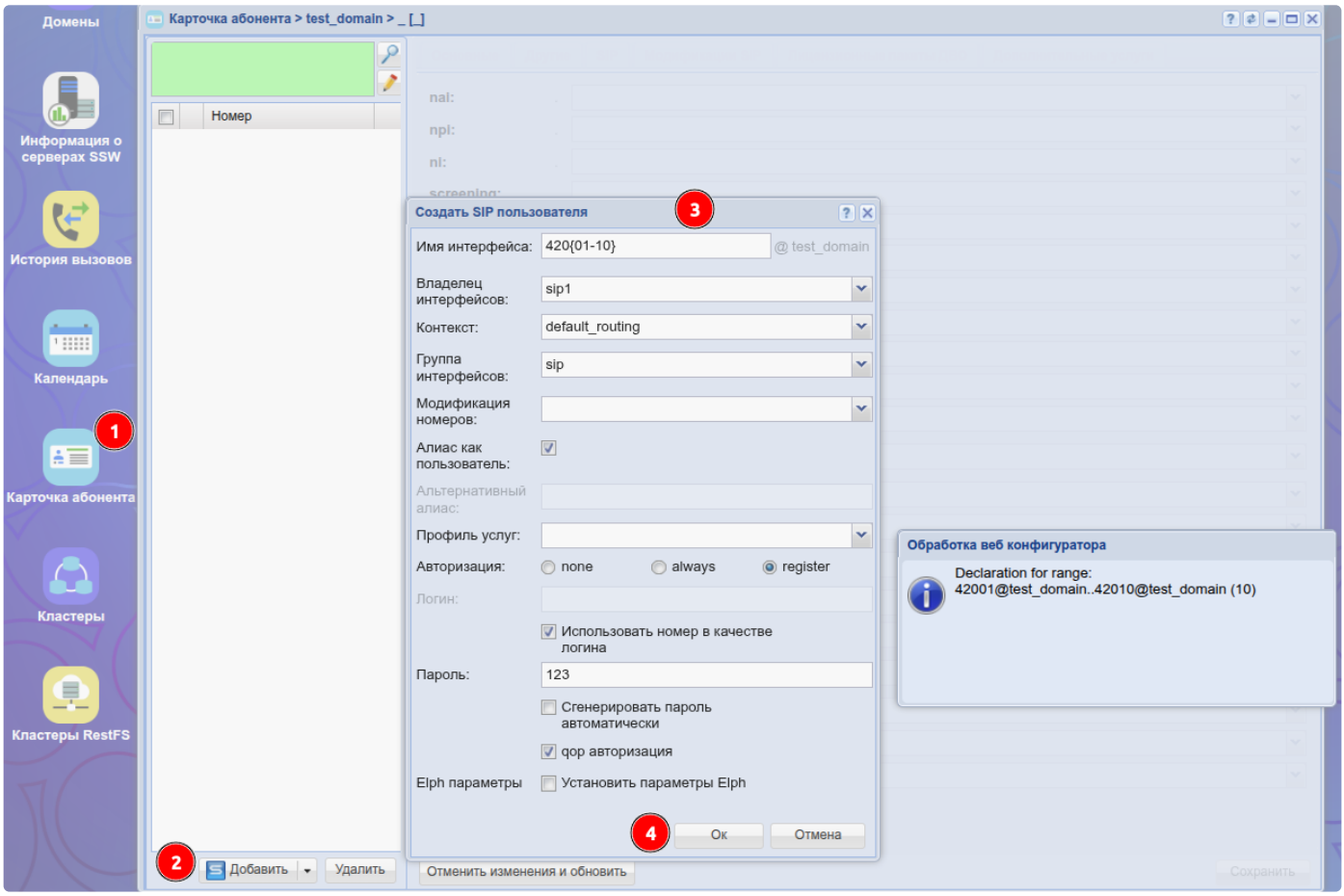


Рисунок 22 — Пример создания 10 абонентов в домене "test_domain"

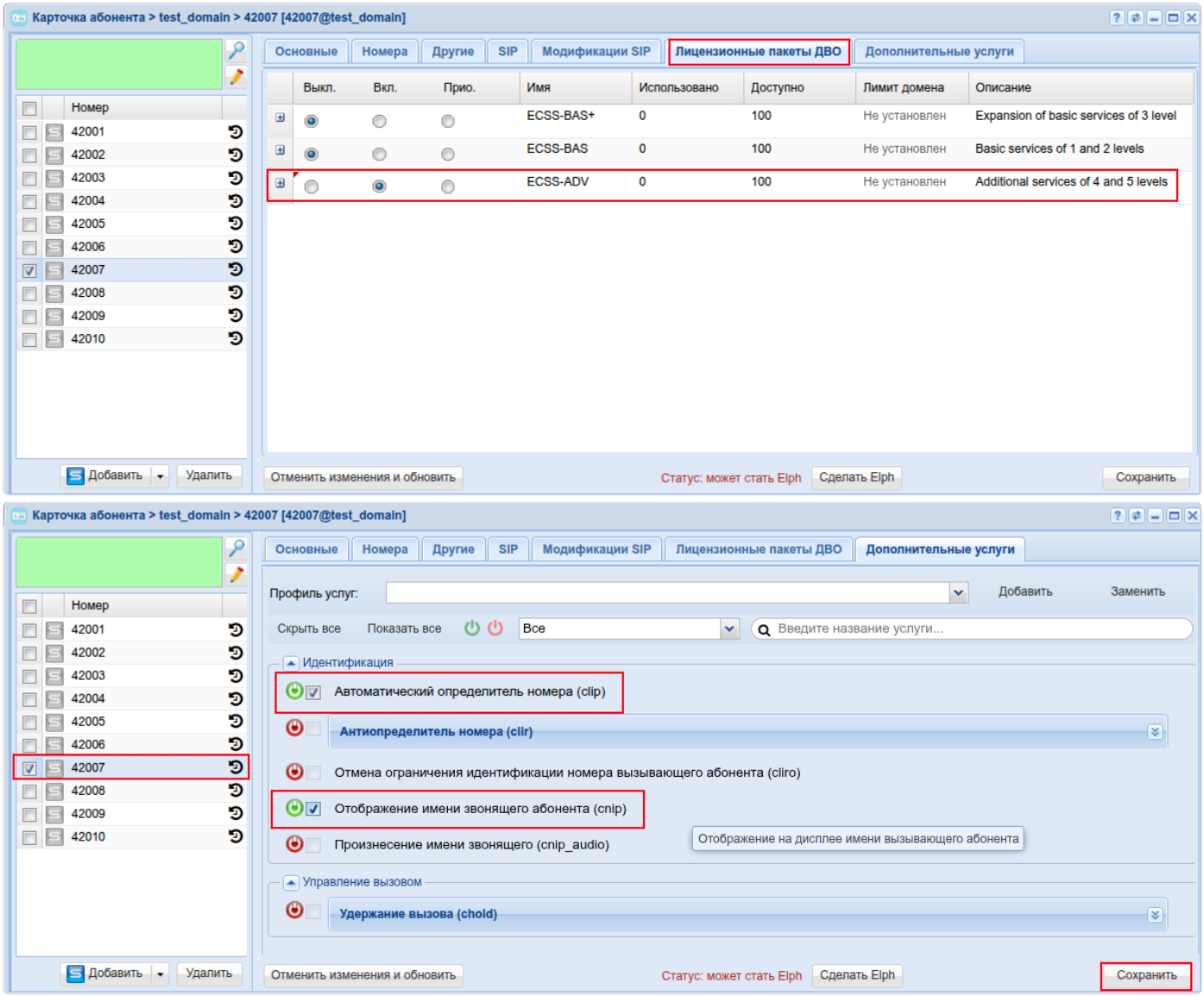


Рисунок 23 — Пример добавления услуг для абонента

13.6.8 Создание и применение контекстов маршрутизации для домена

Маршрутизация отвечает за поиск номера и последующую адресацию вызова. Для корректной работы системы должен быть настроен как минимум один контекст маршрутизации.

Настройка маршрутизации осуществляется в приложении "Менеджер маршрутизации" ("Routing manager") .

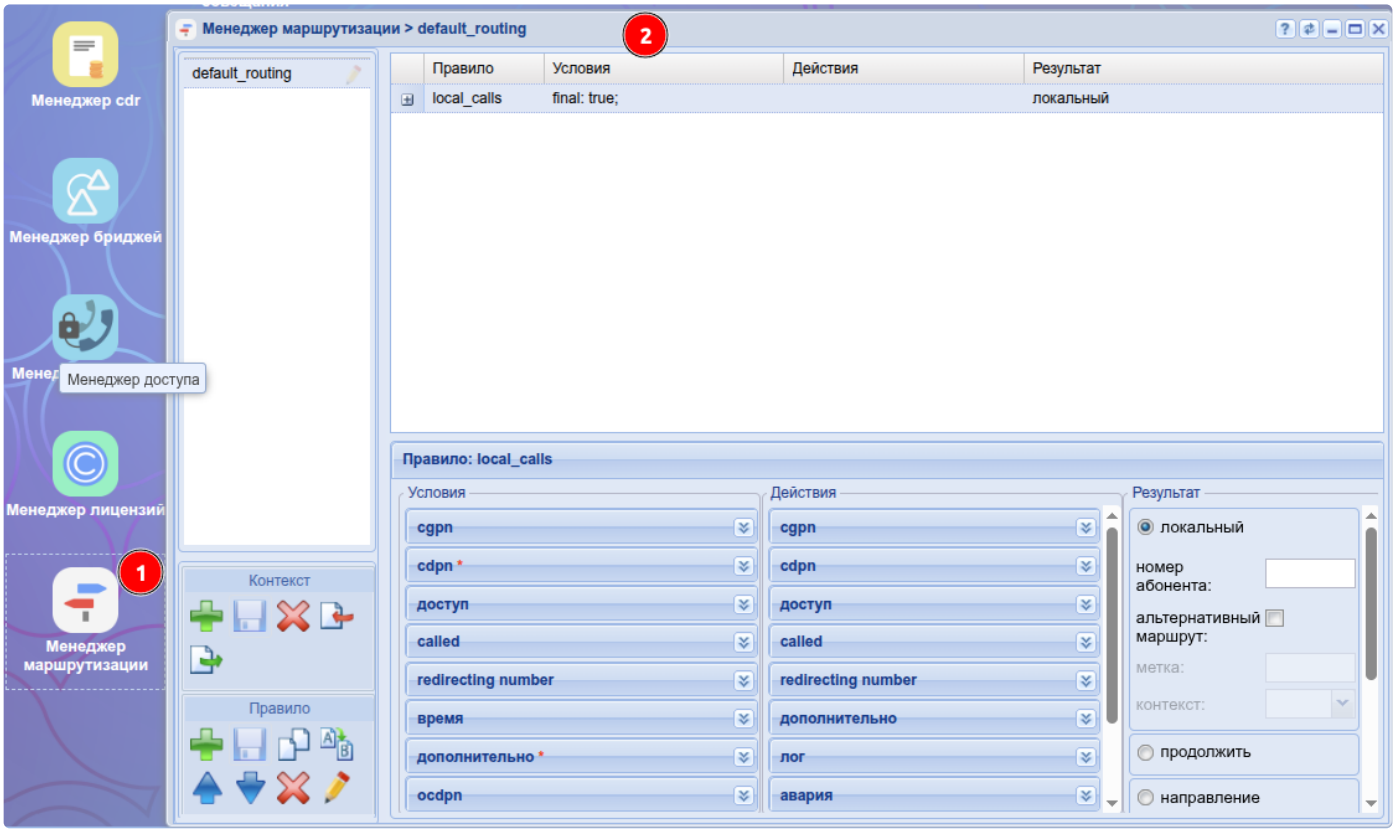


Рисунок 24 — Вид приложения "Менеджер маршрутизации"

В качестве примера создадим контекст, а затем несколько правил в нём, для этого:

1. В левой части окна в разделе "Контекст" нажмите на кнопку *"Создать контекст"* ;
2. В открывшемся диалоговом окне обозначьте имя контекста:

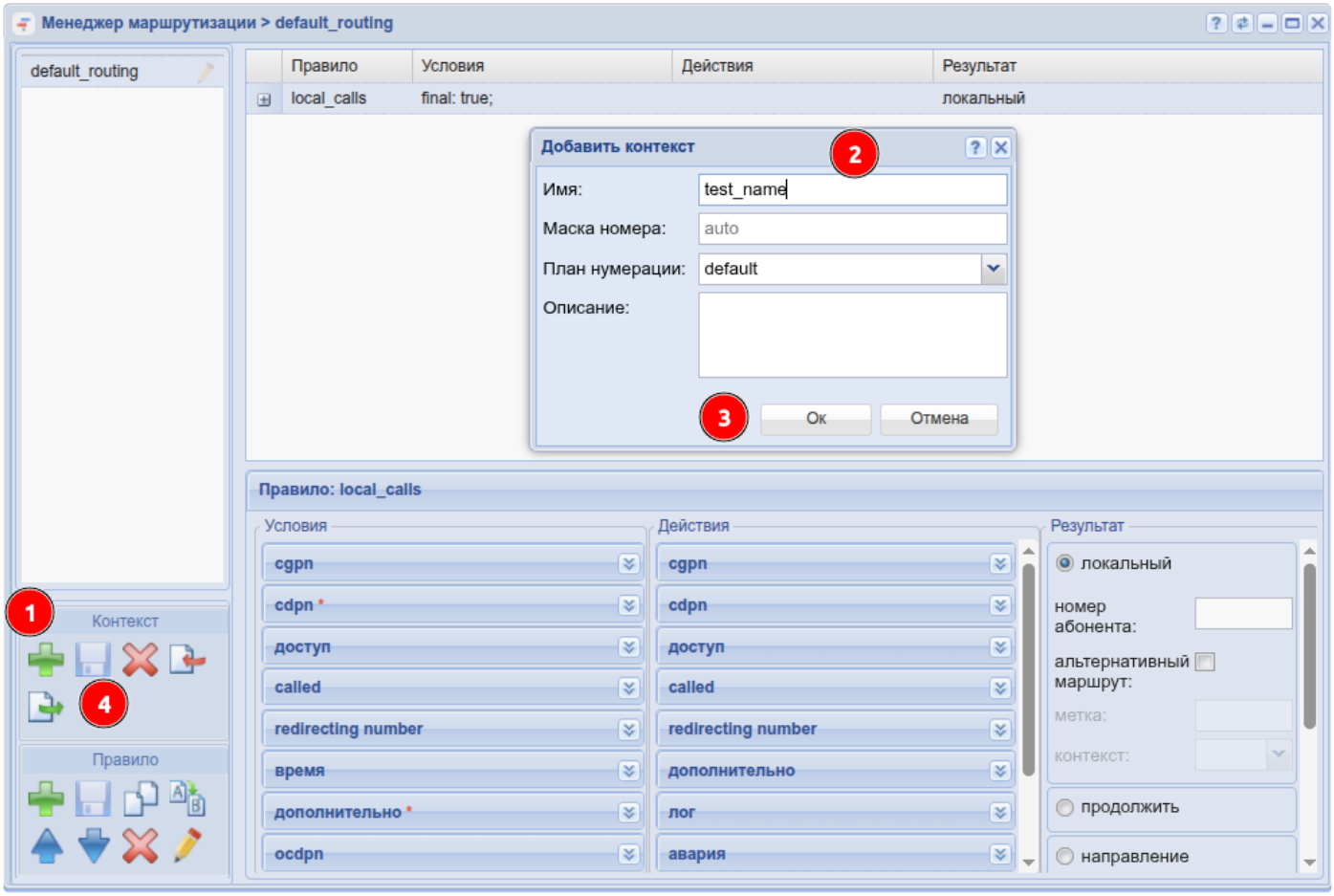


Рисунок 25 — Вид приложения "Менеджер маршрутизации" добавление нового контекста маршрутизации

3. Нажмите *"Сохранить контекст"*  ;

Создадим 4 правила в данном контексте:

- rule1 — правило для выхода на транк TAU-72;
- rule2 — правило локальной маршрутизации для номеров 42001-42005;
- rule3 — правило для выхода в ivr;
- rule4 — правило-исключение.

Для создания нового правила нужно выделить созданный контекст и нажать кнопку *"Создать правило"*. В появившемся окне нужно ввести название правила. Затем сохранить вновь созданные правила.

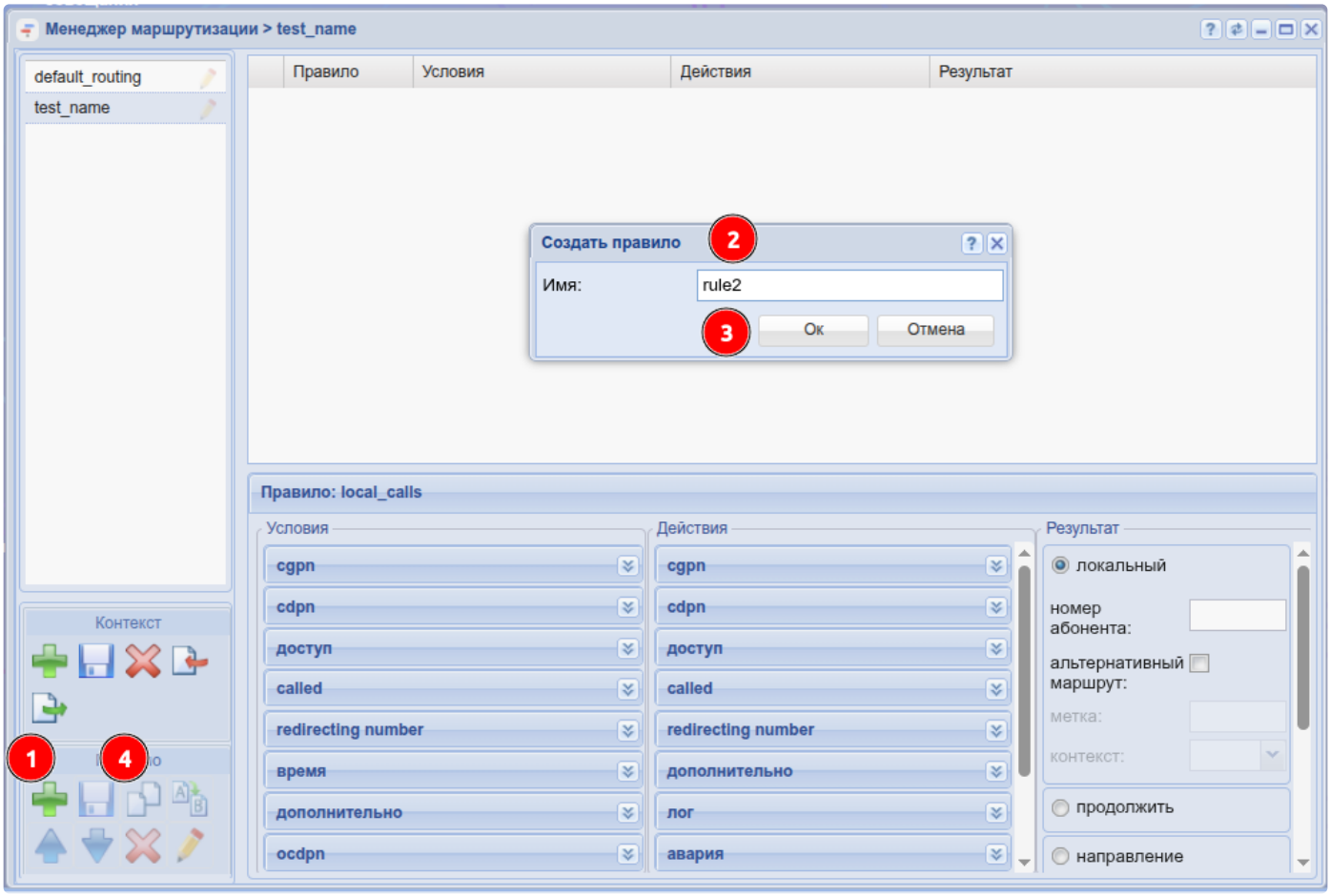


Рисунок 26 – Создание правила контекста маршрутизации

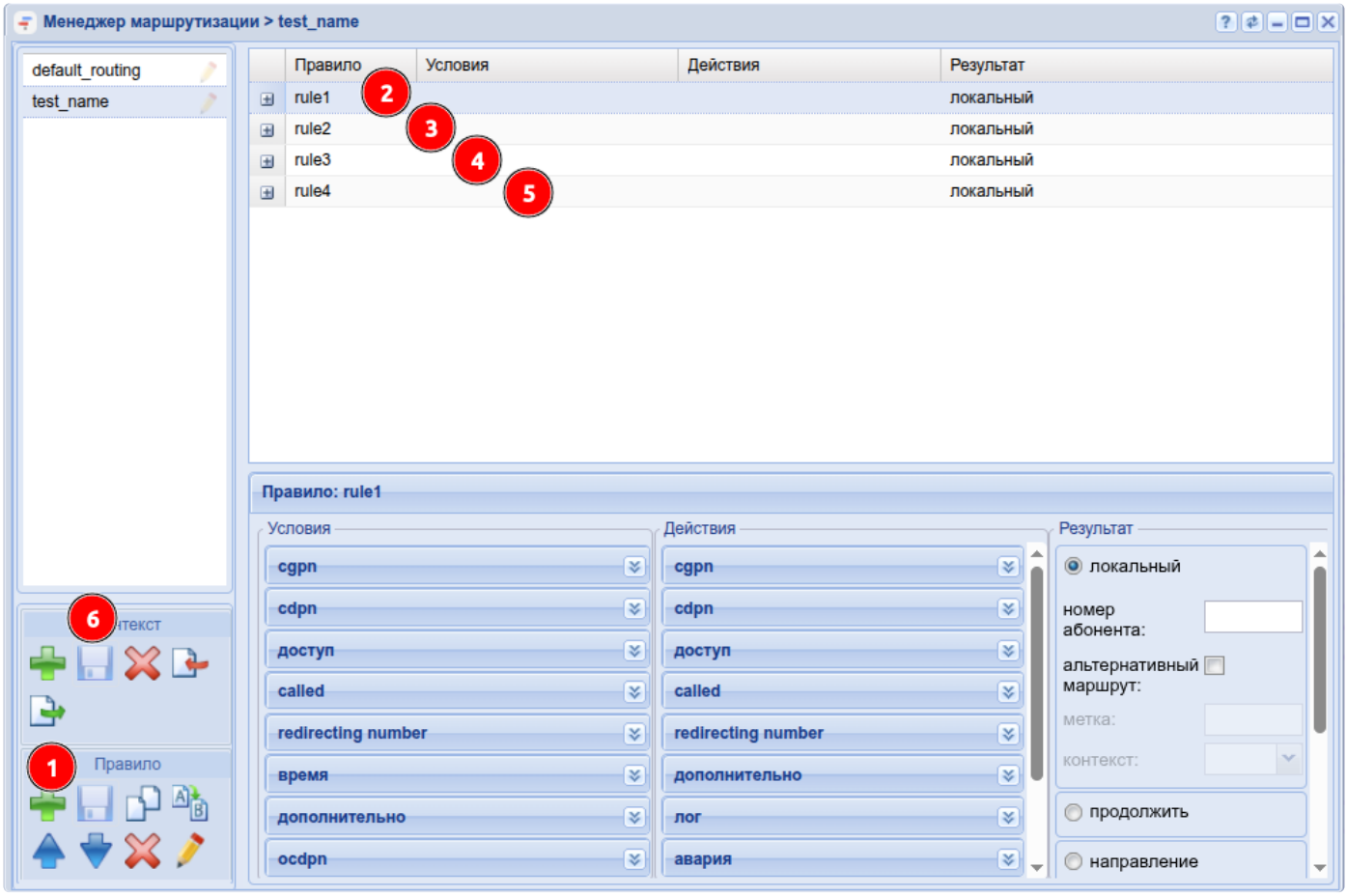


Рисунок 27 — Определение правил

На данный момент не определен транк, чтобы можно было сослаться на него в правиле, однако можно задать номера, по которым будет производится выборка.

Нажав на правило rule1, перейдем в нижнюю часть экрана, в которой располагаются области для редактирования контекста маршрутизации. Условно определим, что выборка для выхода на транк будет проводиться по признакам номера вызываемого абонента (CDPN), а номера в транке должны начинаться с цифры 4.

Функционально контекст маршрутизации делится на три части:

- Условия ("Condition") — раздел, определяющий выражения для выборки по предложенным признакам;
- Действия ("Action") — раздел, выполняющий преобразование признаков номеров к конкретному значению;
- Результат ("Result") — раздел, завершающий маршрутизацию и определяющий её результат.

rule1: Для выхода на транк нужно правильно отредактировать каждую часть:

- В разделе условий перейти на вкладку CDPN. Ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера с 42006 по 42007, нужно ввести условие: 420(6-7);
- В разделе действий перейти на вкладку CDPN. Ввести маску для изменения номера в поле "Номер". Например, чтобы добавить число 4 перед номером, нужно заполнить поле следующим выражением: 4{1,2,3,4,5};
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить транк в системе, поэтому к настройке данного правила вернемся чуть позже.

Таким же образом нужно настроить оставшиеся правила.

Для rule2:

- В разделе условий, во вкладке CDPN ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера со 42001 по 42005, ввести условие 420(1-5);
- В разделе результата определить результат как локальный (т.е. локальная маршрутизация);
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Для rule 3 допустим, что абоненты с номерами 108, 109 и 110 попадают в ivr-скрипт–информатор, прежде чем звонить далее.

- В разделе условий, во вкладке CGPN ввести номера телефонов, по которым будет выборка в поле "Номер", например 420(10,08-09).
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить в системе ivr-скрипт, поэтому к настройке данного правила вернемся чуть позже.

Для rule 4 мы определяем правило-исключение — это правило, которое работает в случае, если любые другие правила не срабатывают.

- По умолчанию данное правило создается в системе в контексте default_routing. В условиях на вызываемый номер прописывается символ %, а результатом является локальная маршрутизация, однако, если мы назначаем абоненту контекст, отличный от default_routing, в конце *рекомендуется* создавать это правило.

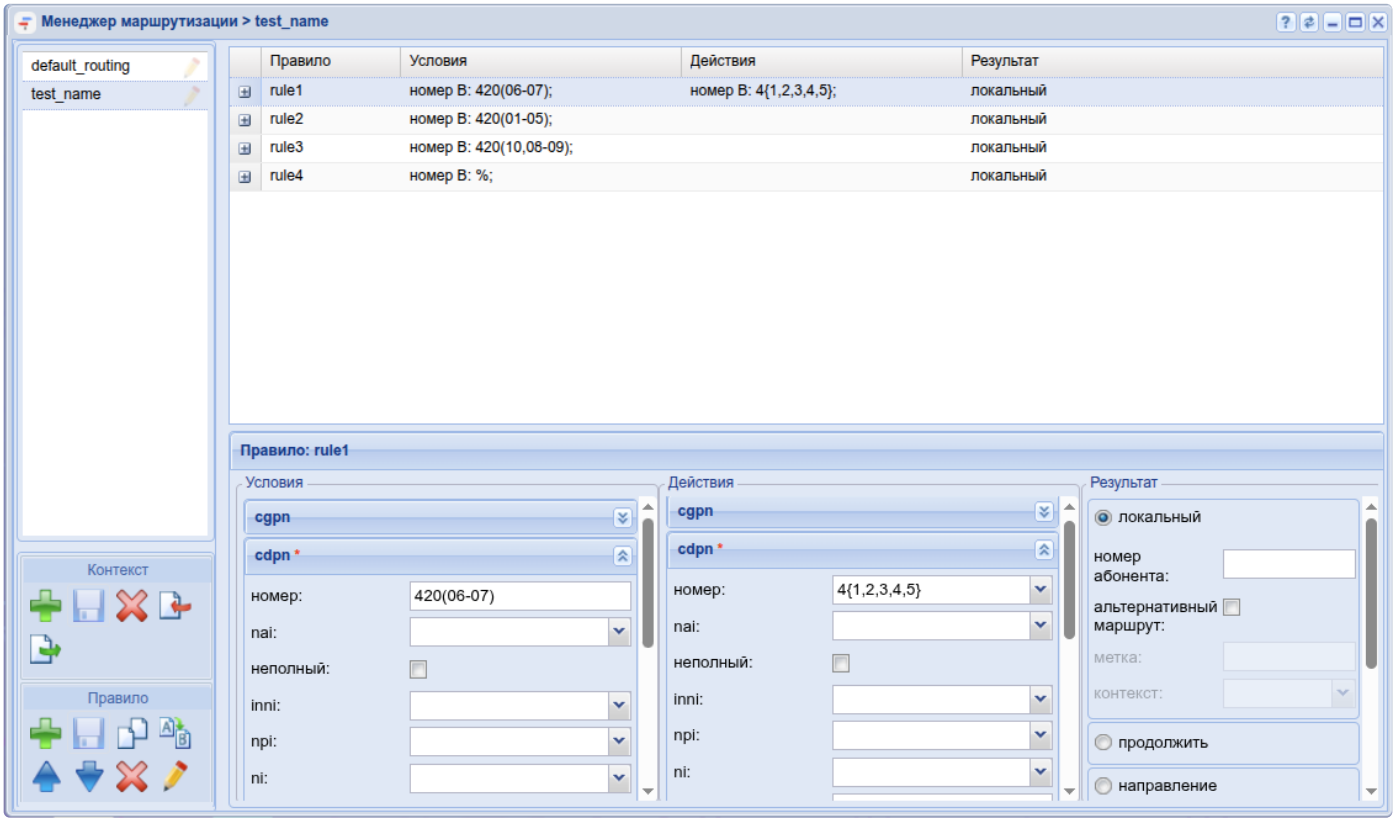


Рисунок 28 — Пример настроек контекста маршрутизации

13.6.9 Создание транка

Для создания и редактирования параметров транка в системе используется приложение "Менеджер транков" ("Trunk manager") .

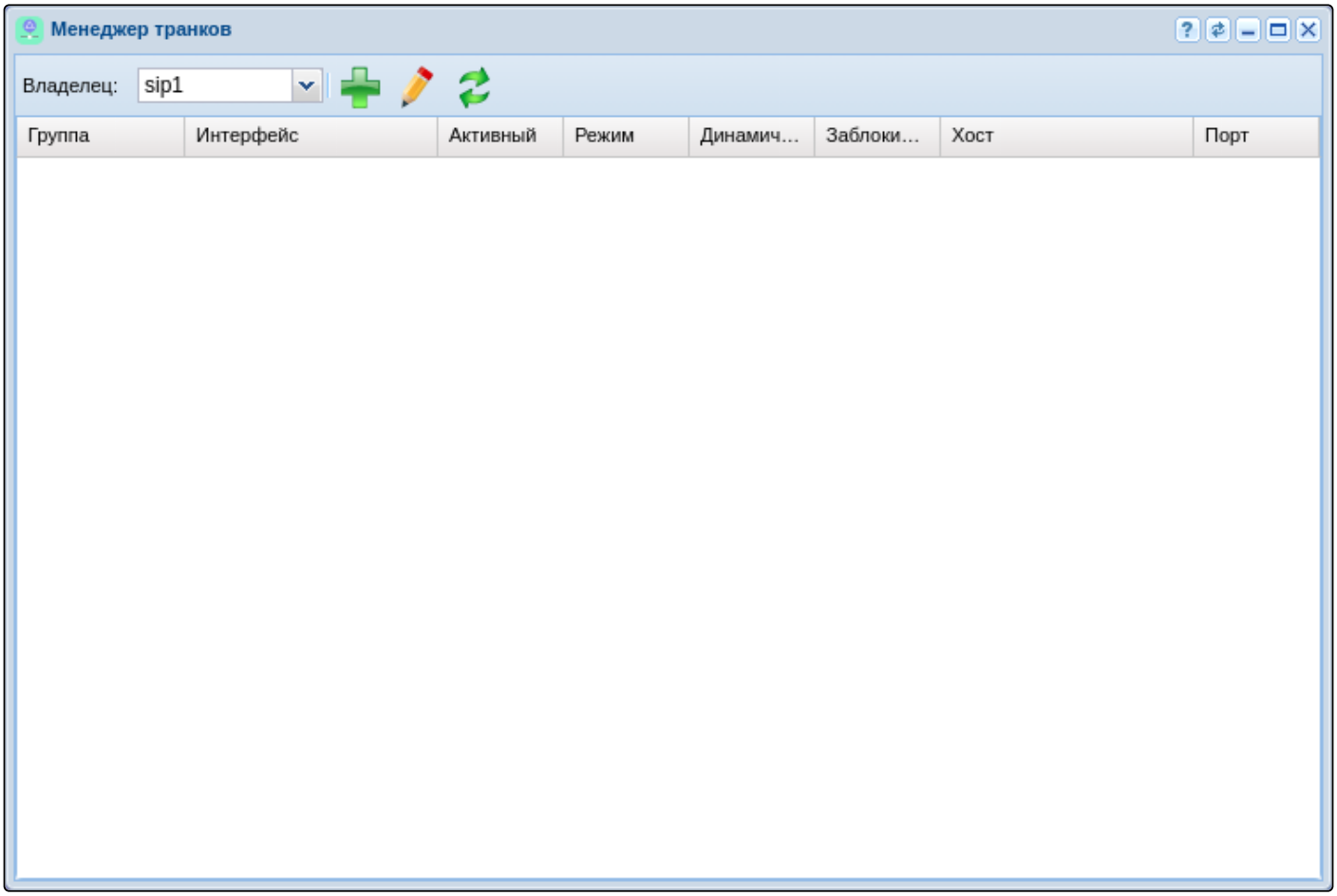


Рисунок 29 — Вид приложения "Менеджер транков"

Для определения транка в системе нужно нажать на кнопку *"Создать транк"* , в появившемся диалоговом окне определить параметры:

- *Имя* — назначить название транка, по которому его можно идентифицировать в системе;
- *Контекст* — применить ранее созданный контекст маршрутизации test_name;
- *Группа* — выбрать группу интерфейсов, созданную при определении абонентов "test.group";
- *Группа IP-адресов (IP-set)* — созданная на домене группа адресов "test_set";
- *Регистрация* — если используется транковая, то нужно включить данный параметр, в примере она не используется;
- *Хост [:порт]* — IP-адрес назначения транка — 10.0.3.100;
- *Порт для приёма* — транспортный порт, на котором будет слушаться трафик с транка, соответствует порту, назначенному на группу IP-адресов.

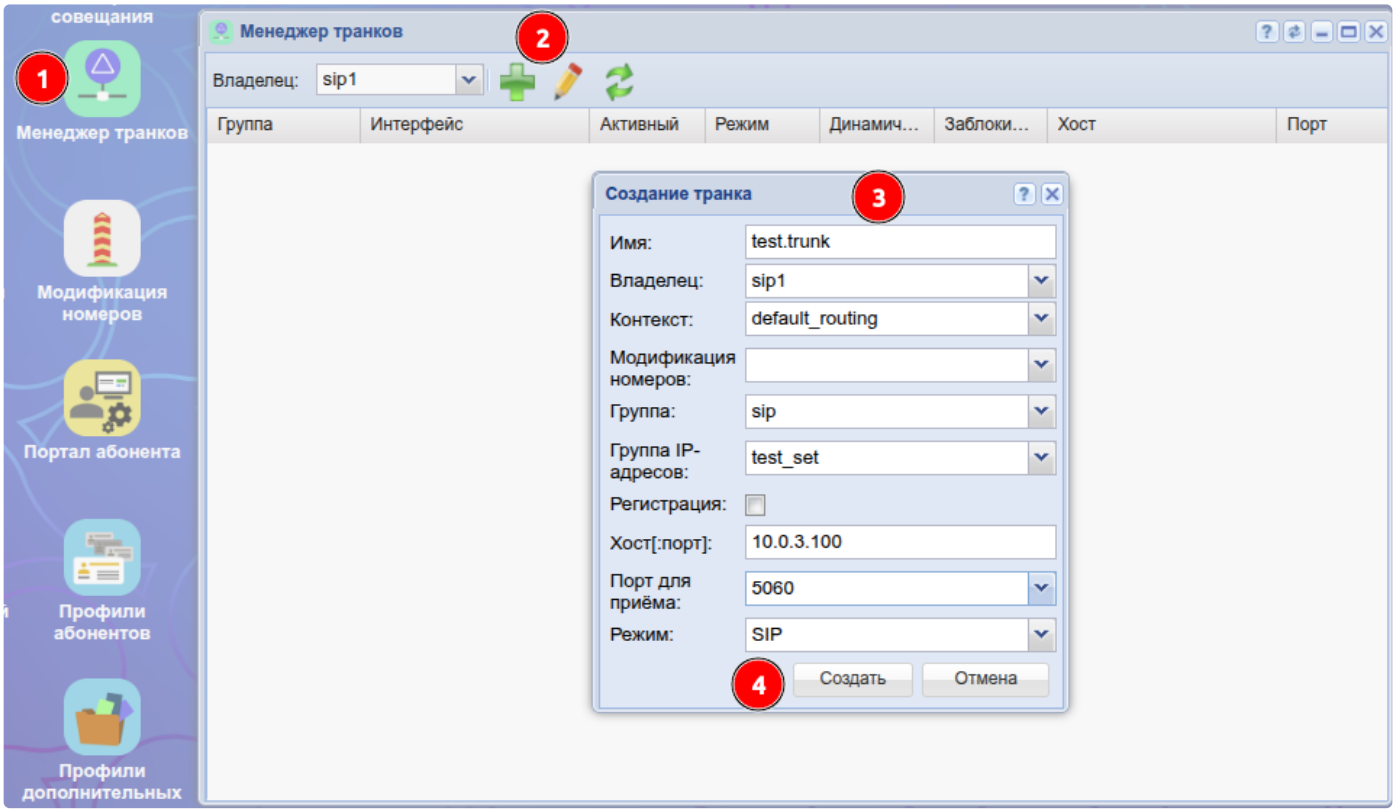


Рисунок 30 — Создание транка

13.6.10 Создание IVR-скрипта

Для создания IVR используется приложение "IVR-редактор" ("IVR editor").

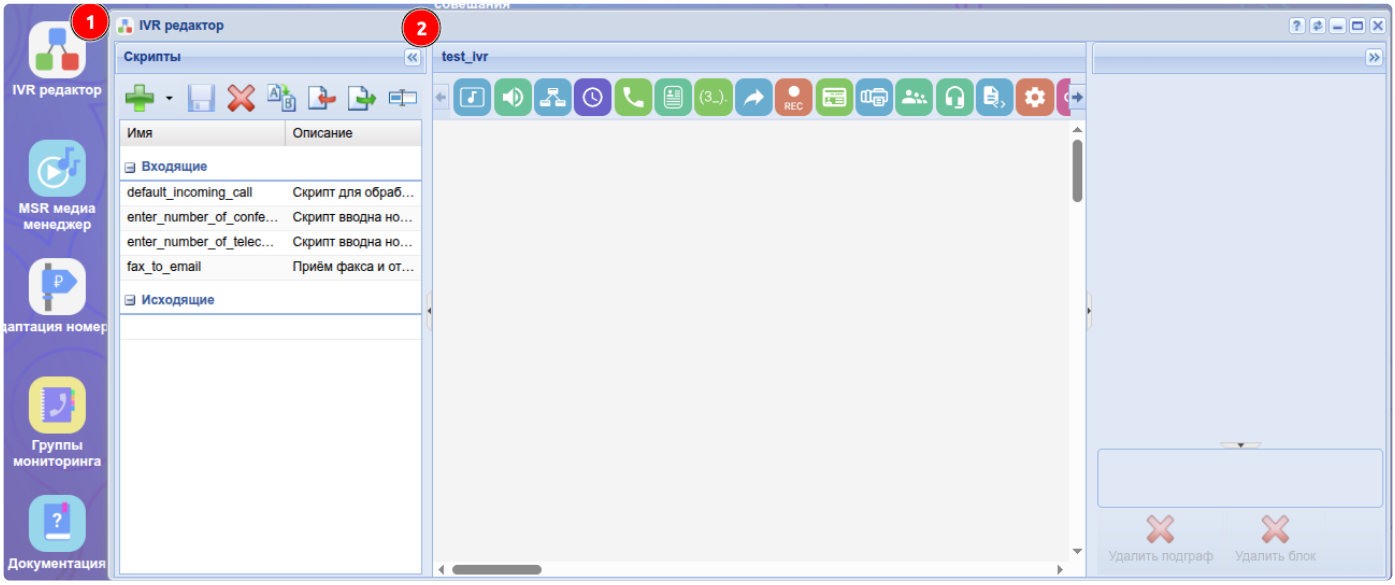


Рисунок 31 — Вид приложения "IVR-редактор"

Для создания скрипта нужно нажать на кнопку *"Добавить"* , выбрать тип скрипта (в данном случае будет скрипт на входящие вызовы), в диалоговом окне указать название скрипта, например "test_ivr".

После создания скрипта в основном окне редактора появится блок-схема.

В качестве примера составим скрипт, который при срабатывании проигрывает вызывающему абоненту заранее записанную фразу, а затем продолжает вызов.

Пример скрипта приведен на рисунках ниже. Здесь используются блоки Info и Dial.

- Блок Info — блок, который проигрывает сообщения до момента ответа пользователя. В качестве записанной фразы используется генератор тонов:

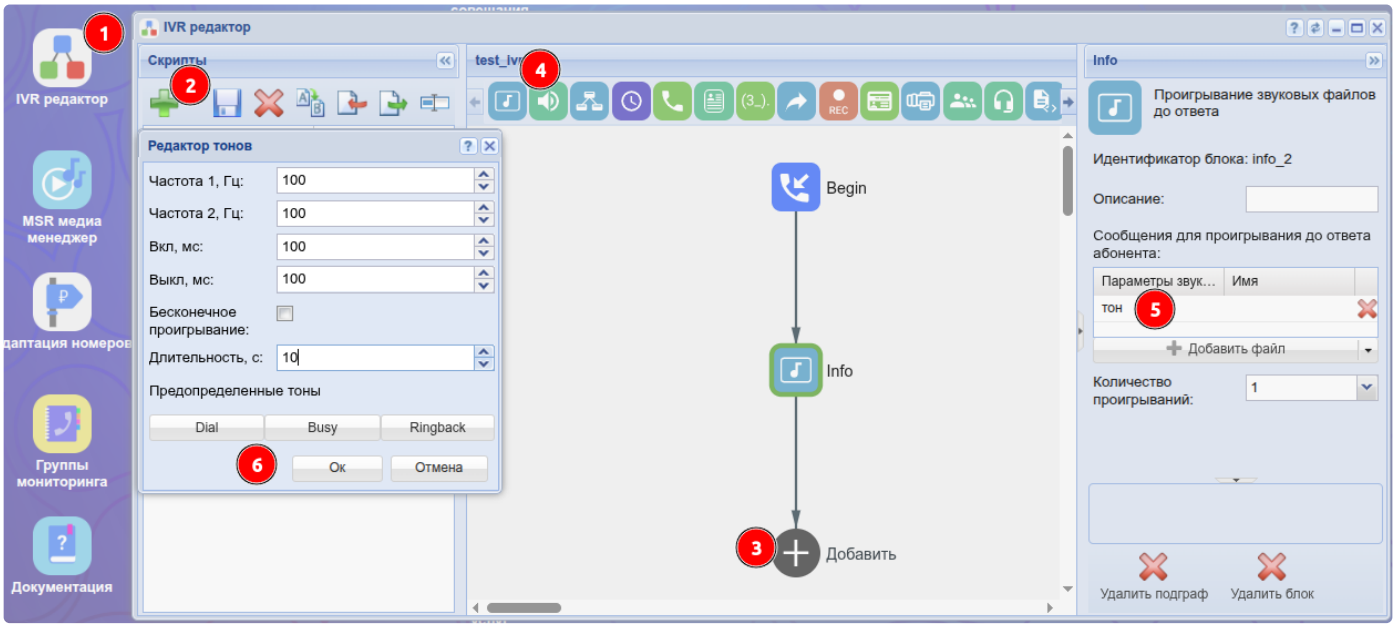


Рисунок 32 — Вид рабочей области IVR с настройками блока "Info"

- Блок dial — блок, который совершает вызов на заданный номер. Чтобы продолжить вызов, нужно применить предопределенную переменную CDPN:

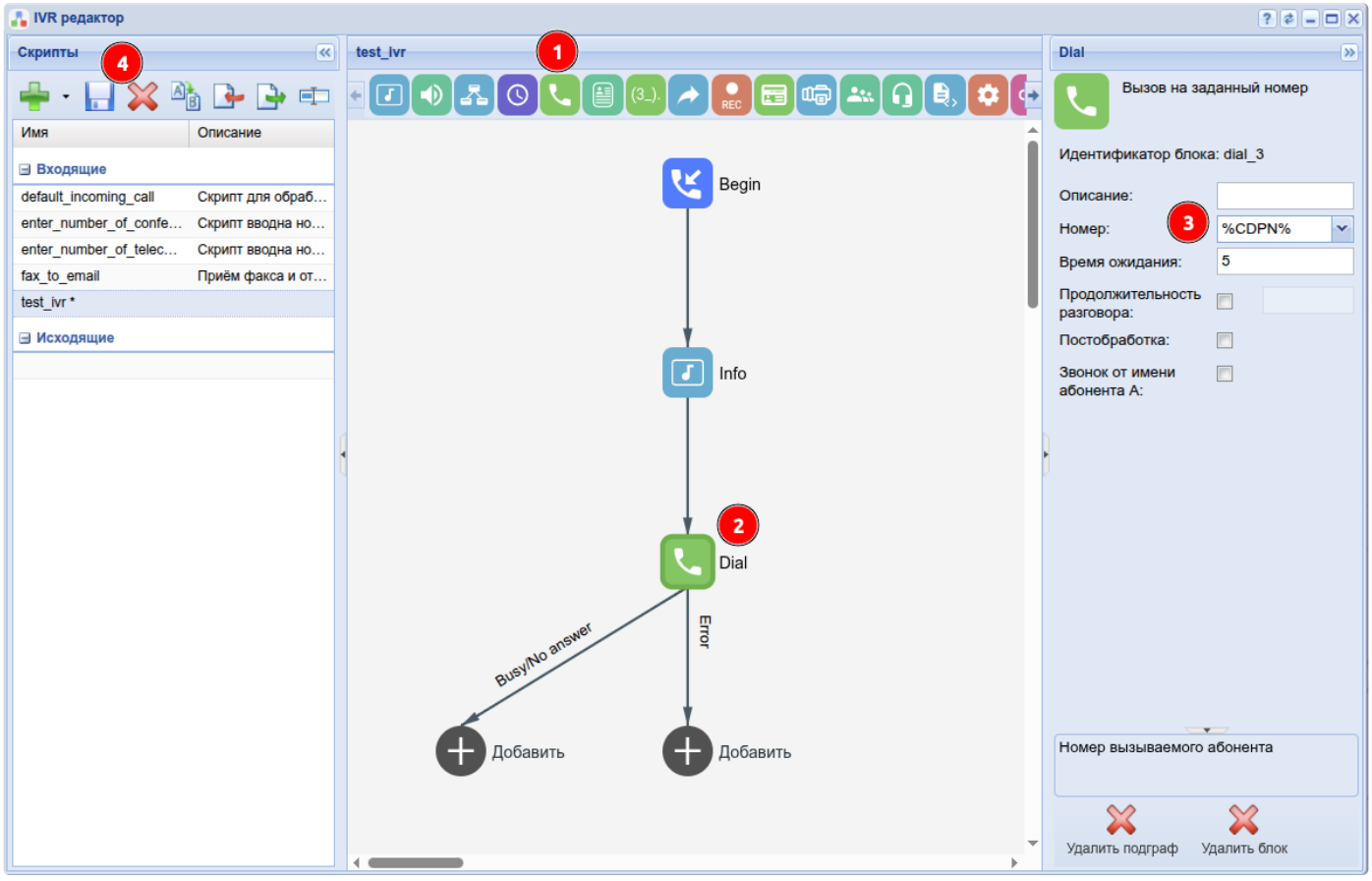


Рисунок 33 — Вид рабочей области IVR

13.6.11 Завершение настройки маршрутизации

Для завершения настройки маршрутизации нужно войти в приложение "Менеджер маршрутизации" ("Routing manager") и в правилах rule1 и rule3 настроить соответствующие результаты маршрутизации, для этого:

- rule1: в разделе "Результат" выберите подпункт "внешний". В таблице "Направления" в поле "Значение" добавьте созданный ранее транк, после чего сохраните правило .
- rule3: В разделе "Результат" выберите подпункт "ivr" и в поле скрипт добавьте созданный ранее ivr-скрипт, после чего сохраните правило и контекст .

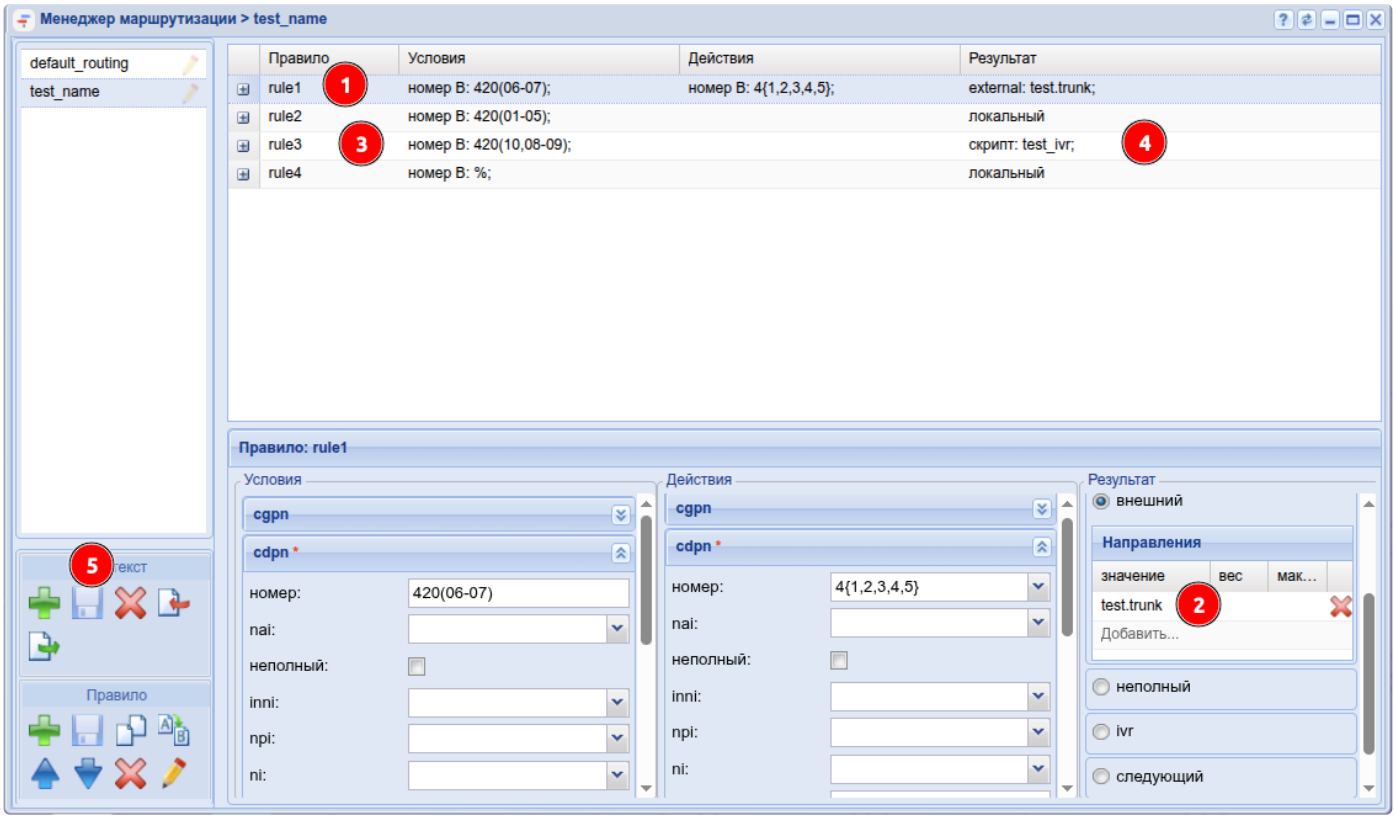


Рисунок 34 — Вид контекста маршрутизации

13.7 Первоначальная конфигурация системы через интерфейс командной строки (CLI)

пример первоначальной конфигурации через CoCon (CLI)

13.7.1 Добавление медиа-ресурсов в интерфейсе командной строки (CLI)

⚠ Для возможности использования медиаресурса и установления соединений , необходимо выполнить активизацию в команде создания медиаресурса - **/system/media/resource/declare** (параметр **"is active"=true**) или **/system/media/resource/set** (параметр **"active"=true**), если при создании активизация не была выполнена. Так же можно выполнить активизацию через веб интерфейс , приложение - сетевые окончания MSR (MSR-registrars).

Зайдите в CLI - CoCon:

```
ssh admin@localhost -p 8023
password: password
```

Медиаресурс — это описание параметров медиасервера, необходимых для работы с ним.

Медиаресурс — это описание параметров медиасервера, необходимых для работы с ним.

declare
Команда для объявления подключения медиасервера к ECSS-10.

Для работы в режиме предварительно настроенной системы можно заранее прописать медиаресурсы, информация о которых известна.

Для этого необходимо создать описание известных медиасерверов с указанием всех параметров, что позволить начать работу с медиасервером сразу после его регистрации.

Команда для создания конфигурации медиасервера:

Путь команды:

/system/media/resource/declare

Синтаксис:

declare <node> by_contact <contact> <zone> <site> <is active>

или

declare <node> by_iface <msr> <acc_id> <iface> <zone> <site> <is active>

Параметры:

<node> — имя ноды core;
<contact> — contact uri декларируемого медиаресурса;
<zone> — логическая зона обслуживания MSR;
<site> — идентификатор сети обслуживания MSR;
<is active> — состояние активности медиаресурса, значения true/false;
<msr> — имя MSR;
<acc_id> — номер идентификатор ресурса;
<iface> — название интерфейса.

Установить связь MSR- и Core-подсистем. Для этого выполнить команды:

```
/system/media/resource/declare core1@ecss1 contact net.20@msr.ecss1  default local true
```

результат

```
/system/media/resource/info core1@ecss1 net.20@msr.ecss1

  Media resource specific for node core1@ecss1:
┌─┬──────────────────┬──────────────────────┐
│F│    Parameter     │        Value         │
├─┼──────────────────┼──────────────────────┤
│r│ Contact          │ net.20@msr.ecss1     │
```

```
|r| Status           | registered           |
|r| Cc-status        | connected            |
|r| Cc-status uptime| 00:29:06             |
|r| Cc-id            | 63bfc9               |
|r| Address          | 10.0.20.40:5700      |
|r| Register time    | 2025/12/02 13:38:58 |
|r| Expired          | 113                  |
|r| Iface            | net.20(10.0.20.40)   |
| | Active           | true                 |
| | Zone             | default              |
| | Site             | local                |
|r| MSR version      | 3.18.1.0.10          |
|r| MSR load         | 0                    |
|r| MSR perf coef    | 1.0                  |
└─┴──────────────────┴──────────────────────┘

Legend:
Column F - means Flag, values r - flag for readonly parameter
```

результат / список

```
/system/media/resource/list all
  All media resource selected list specific:
┌──────────────┬──────────┬──────────────┬──────────┬──────┬────────┬──────────┬──────────
──┬──────────┬──────────┐
│    Node      │    MSR   │     MSR      │    MSR   │ MSR  │ Cc-id  │ Cc-status │ Cc-uptime
│   Cc-address │ Iface    │   Iface      │ Active   │ Zone │ Site   │    Contact    │
   Status  │ Expired │
│              │          │   version    │ perf coef │ load │        │           │
│              │  name    │    addr      │          │      │        │           │
│         │
├──────────────┼──────────┼──────────────┼──────────┼──────┼────────┼──────────┼──────────
──┼──────────┼──────────┤
│ core1@ecss1 │ msr.ecss1 │ 3.18.1.0.10 │      1.0 │    0 │ 63bfc9 │ connected │ 00:27:50
│ 10.0.20.40:5700 │ net.20 │ 10.0.20.40 │ true    │ default │ local │ net.20@msr.ecss1 │
 registered │      68 │
└──────────────┴──────────┴──────────────┴──────────┴──────┴────────┴──────────┴──────────
──┴──────────┴──────────┘
```

13.7.2 Добавление управления регистратором для MSR медиа менеджера в интерфейсе командной строки (CLI)

/system/media/registrar/ - команды управления регистратором для MSR медиа менеджера

/system/media/registrar/ - команды управления регистратором для MSR медиа менеджера

set

Команда для изменения общих и индивидуальных параметров.

Путь команды:

/system/media/registrar/set

Синтаксис настройки общих параметров:

set common <FIELD> <VALUE>

Параметры:

<FIELD> - имя настраиваемого параметра;

<VALUE> - значение настраиваемого параметра

Где

- common - тип выставляемого параметра;
- <common parameter> - общий параметр медиарегистратора для всех нод: registration-timeout;
- private - тип выставляемого параметра;
- <node name> - имя ноды на которой находится медиарегистратор;
- <private parameter> - параметр медиарегистратора, который выставляется на <node name>: is-active, listen-interface & listen-port.

Список параметров доступных для настройки:

- is-active - позволяет включить или выключить регистратор (false - деактивирует регистратор, регистратор перестает принимать SIP трафик);
- listen-interface - имя интерфейса, который использует регистратор;
- listen-port - порт, на котором регистратор принимает запросы на регистрацию (по умолчанию 5000);
- registration-timeout - таймер перерегистрации, диапазон значений от 60 до 86400.

В общих параметрах доступна настройка времени регистрации, по умолчанию установлено 60 секунд.

Пример:
Установить время регистрации 60 секунд:

```
/system/media/registrar/set common registration-timeout 60
Common parameter registration-timeout set to 60.
```

Синтаксис настройки индивидуальных параметров:

system/media/registrar/set private <CORE> <FIELD> <VALUE>

Параметры:

<CORE> - нода ядра, для которой настраивается Registrar-сервер;

<FIELD> - имя настраиваемого параметра;

<VALUE> - значение настраиваемого параметра.

Настраиваемые параметры:

- is-active - активен ли Registrar-сервер, принимает значения true | false;
- listen-ip - ip-адрес, на котором Registrar-сервер будет принимать соединения (по умолчанию 0.0.0.0);
- listen-port - сетевой порт, на котором Registrar-сервер будет принимать соединения (по умолчанию 5000).

Установить ipадрес для приема соединений от медиасервера:

```
/system/media/registrar/set private core1@ecss1 listen-ip 10.0.20.40
Private parameter listen-ip set to 10.0.20.40 at node core1@ecss1.
```

Результат:

```
/system/media/registrar/info
 Common parameters:
┌──────────────────────────────┬─────────┐
│          Parameter           │  Value  │
├──────────────────────────────┼─────────┤
│ registration-timeout, sec    │   180   │
└──────────────────────────────┴─────────┘
 Private parameters:
┌──────────────┬──────────┬────────────┬─────────────┬──────────────┐
│     Node     │  Status  │ Is-active  │  Listen-ip  │ Listen-port  │
├──────────────┼──────────┼────────────┼─────────────┼──────────────┤
│ core1@ecss1  │ active   │ true       │ 10.0.20.40  │         5000 │
└──────────────┴──────────┴────────────┴─────────────┴──────────────┘
```

После того как все сервисы будут запущены, ноды какое-то время будут устанавливать связь. Как только все ноды загрузятся , при выполнении запроса **system-status**, последует вывод следующей информации:

```
/system-status
Checking...
┌─┬──────────────┬──────────────────────────────┬──────────────┬─────────────┬────────┐
│ │     Node     │           Release            │ Erlang nodes │Mnesia nodes │Uptime  │
├─┼──────────────┼──────────────────────────────┼──────────────┼─────────────┼────────┤
│ │core1@ecss1   │ecss-core-3.18.1.0.1655       │core1@ecss1   │not running  │31m 25s │
│ │ds1@ecss1     │ecss-ds-3.18.1.0.1655         │ds1@ecss1     │ds1@ecss1    │31m 25s │
│ │md1@ecss1     │ecss-mediator-3.18.1.0.1655   │md1@ecss1     │md1@ecss1    │31m 26s │
│ │mycelium1@ecss1│ecss-mycelium-3.18.1.0.1655  │mycelium1@ecss1│not running  │31m 26s │
│ │sip1@ecss1    │ecss-pa-sip-3.18.1.0.1655     │sip1@ecss1    │sip1@ecss1   │31m 26s │
└─┴──────────────┴──────────────────────────────┴──────────────┴─────────────┴────────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────┬────────────┬──────────────┐
│     Node     │   MSR    │   MSR    │ Cc-status  │ Cc-uptime    │
│              │          │ version  │            │              │
├──────────────┼──────────┼──────────┼────────────┼──────────────┤
│ core1@ecss1  │ msr.ecss1 │ 3.18.1.0.10 │ connected │ 00:29:51   │
└──────────────┴──────────┴──────────┴────────────┴──────────────┘
```

В примере выше видно, что ноды вошли в кластер и *MSR* зарегистрировались на ноде *ecss-core*.

Создание и настройка группы SIP-IP-адресов (IP-set)

IP-set – совокупность одного или нескольких IP-адресов из одной подсети и списка слушающих UDP/TCP-портов, которые будут открыты SIP-адаптером на данных адресах.

Произведите настройку адаптера SIP согласно ТЗ (команды выполнить в CoCon):

```
/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss1 ip =
10.0.20.40
Property "ip_set" successfully changed from:

to
test_set: no ports set
test_set: sip1@ecss1 10.0.20.40
test_set: dscp 0.
```

```
/cluster/adapter/sip1/sip/network/set ip_set test_set listen-ports list = [5060]
Property "ip_set" successfully changed from:
test_set: no ports set
test_set: sip1@ecss1 10.0.20.40
test_set: dscp 0
   to
test_set: 5060
test_set: sip1@ecss1 10.0.20.40
test_set: dscp 0
```

13.7.3 Создание Домена и подключение IP-set

Создайте Домен (виртуальную АТС) и присвойте ему созданную выше группу (IP-set) настроек SIP-адаптера (команды выполняются в CoCon):

```
/domain/declare test_domain --add-domain-admin-privileges --add-domain-user-privileges
New domain test_domain is declared
```

```
/domain/test_domain/sip/network/set ip_set [test_set]

Property "ip_set" successfully changed from:

[]
    to
["test_set"].
```

После создания домена, администратор системы должен открыть доступ на управление услугами оператору Домена (виртуальной АТС). Для этого используются команду:

Разрешить оператору Домена (виртуальной АТС) "test_domain" управлять всеми услугами :

```
/cluster/storage/ds1/ss/access-list add test_domain *
```

```
Supplementary services successfully added in the domain test_domain.
```

Для возможности применения пакета услуг у абонентов сразу после создания, необходимо настроить политику применения лицензионных пакетов услуг (детальную информацию вы можете посмотреть здесь)

Создать политику применения лицензии с именем -"new_sub"

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/declare new_sub "Задаются при создании абонента"
```

Задать имя пакета услуг дополнительных сервисов, для этого использовать следующие команды :

выполняется в CoCon user/pw admin/password

```
/cluster/storage/ds1/licence/show-licence 1
```

нас интересует следующая информация :

```
/cluster/storage/ds1/licence/show-licence 1
SSW ID                                                  < Имя SSW ID >
Description                                             Лицензия для тестирования.

. . .

Supplementary Services licence package(s)
 name                                                   'ECSS-ADV'
 limit                                                  100
 description                                            "Additional services of 4 and
5 levels"
 SS list                                                [1,2,3,4,5]

. . .
```

имя пакета услуг в примере "ECSS-ADV".
Добавить в созданную политику пакет лицензионных услуг с именем "ECSS-ADV", именно так он описан в лицензии:

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/package-add new_sub ECSS-ADV
```

Пакет добавлен, но в настоящий момент политика не активна (не может быть применена к абоненту). Для активации политики new_sub выполнить следующую команду:

выполняется в CoCon user/pw admin/password

```
/domain/test_domain/ss/licence/politics/activate new_sub
```

Теперь при создании абонентов можем использовать дополнительные услуги.

14 Пример первичной конфигурации системы с помощью web-конфигуратора

- Исходные данные
- Подготовка к работе
- Создание учетной записи оператора
- Создание домена
- Создание IP-set (sip-транспорта) и присвоение его к домену
- Создание абонентов
- Создание и применение контекстов маршрутизации для домена
- Создание транка
- Создание IVR-скрипта
- Завершение настройки маршрутизации
- Настройка услуг
- Добавление медиа-ресурсов.

14.1 Исходные данные

- Установка системы завершена;
- Система готова к дальнейшей настройке;
- Интерфейсы запущены.

Рекомендуется использовать последние доступные версии браузеров. Рекомендуемые к использованию браузеры: Opera, Chrome.

Для начала конфигурации системы необходимо зайти в web-конфигуратор.

Для определения и регистрации в системе планируются:

- Абоненты с номерами 101, 102 , 103, 104 ,105, 106, 107, 108, 109, 110;
- Транк в сторону шлюза.

14.2 Подготовка к работе

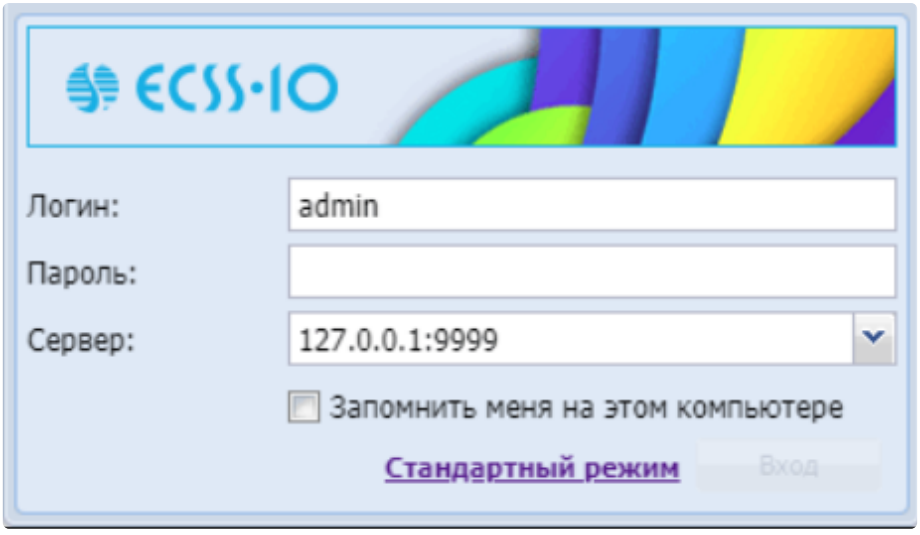


Рисунок 3 — Вход в web-конфигуратор (окно авторизации)

В окне авторизации необходимо ввести значения, определенные при установке web-конфигуратора.

⚠ Стандартные значения для авторизации:
Логин: admin
Пароль: password

После входа в систему будет видна основная рабочая область с иконками приложений, а также строка состояния с доступными опциями, в частности:

- 1 — выход из системы;
- 2 — выбор домена;
- 3 — выбор языка.



Рисунок 4 — Вид рабочей области web-конфигуратора

14.3 Создание учетной записи оператора

После авторизации для повышения безопасности при эксплуатации программного коммутатора рекомендуется создание учетных записей для операторов, а также сменить пароль для пользователя **admin**.

Для создания новой учетной записи оператора используется приложение "Управление пользователями" ("User manager"):

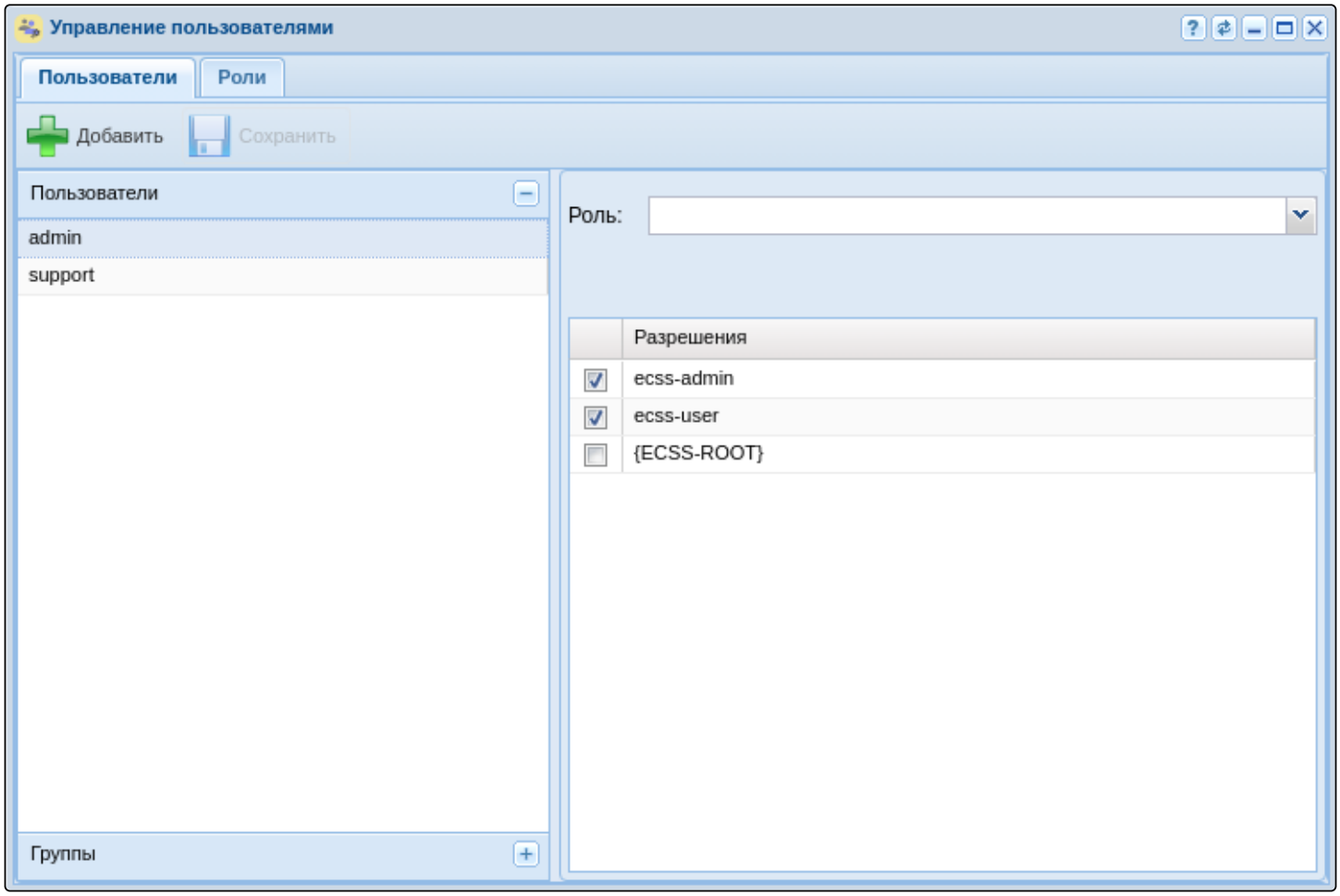


Рисунок 5 — Вид приложения "*Управление пользователями*"

Нажмите кнопку "*Добавить*"  . В открывшемся окне требуется определить новую учетную запись, для этого:

1. В поле "*Имя*" введите логин учетной записи, например "test";
2. В поле "*Пароль*" и "*Подтверждение*" введите пароль для пользователя, например "testpassword";
3. Определите уровень прав доступа для пользователя через выбор текущих разрешений или с помощью ролей, например **ecss-user**. Описание можно посмотреть в разделе "Управление пользователями".

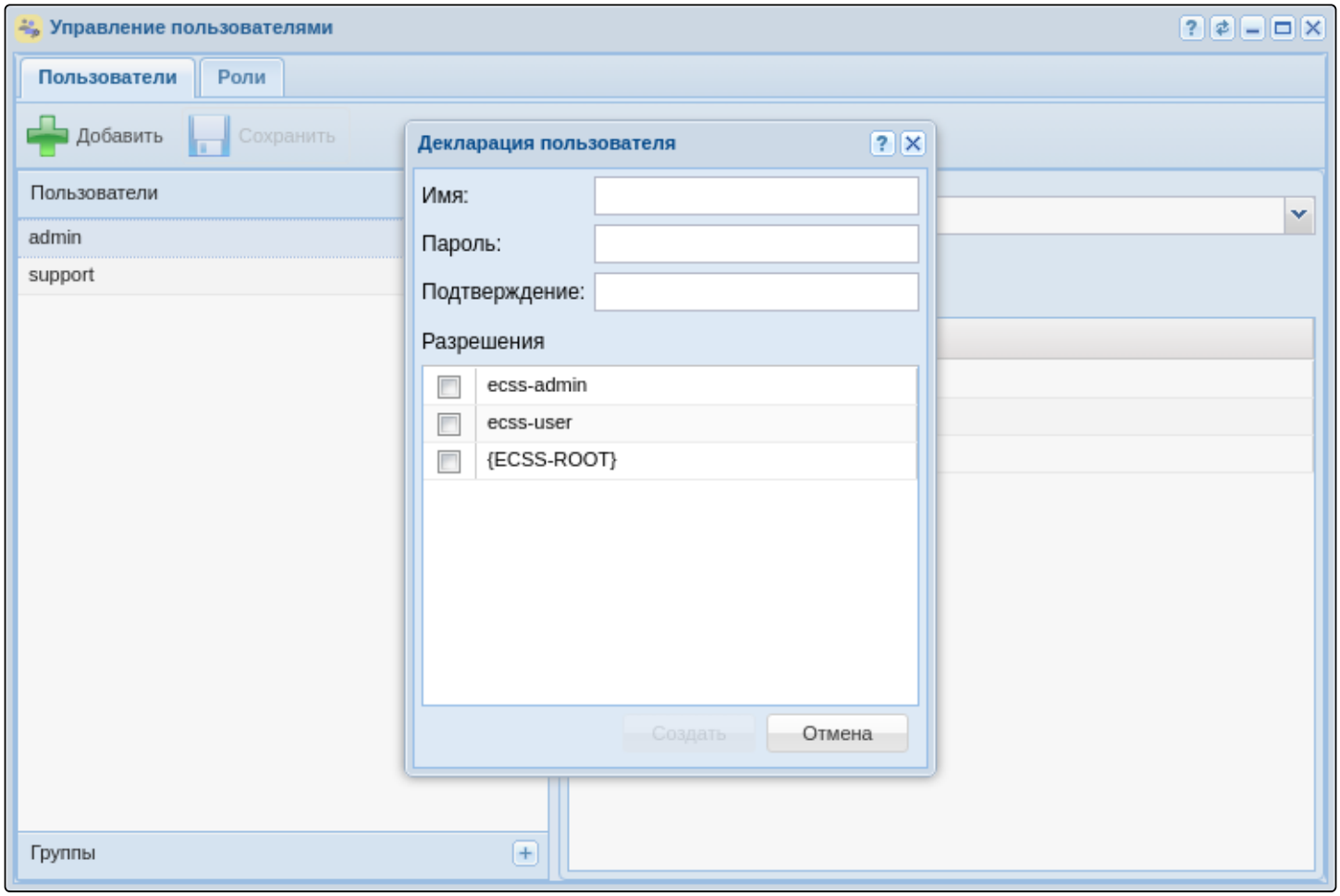


Рисунок 6 – Диалоговое окно создания учетной записи оператора

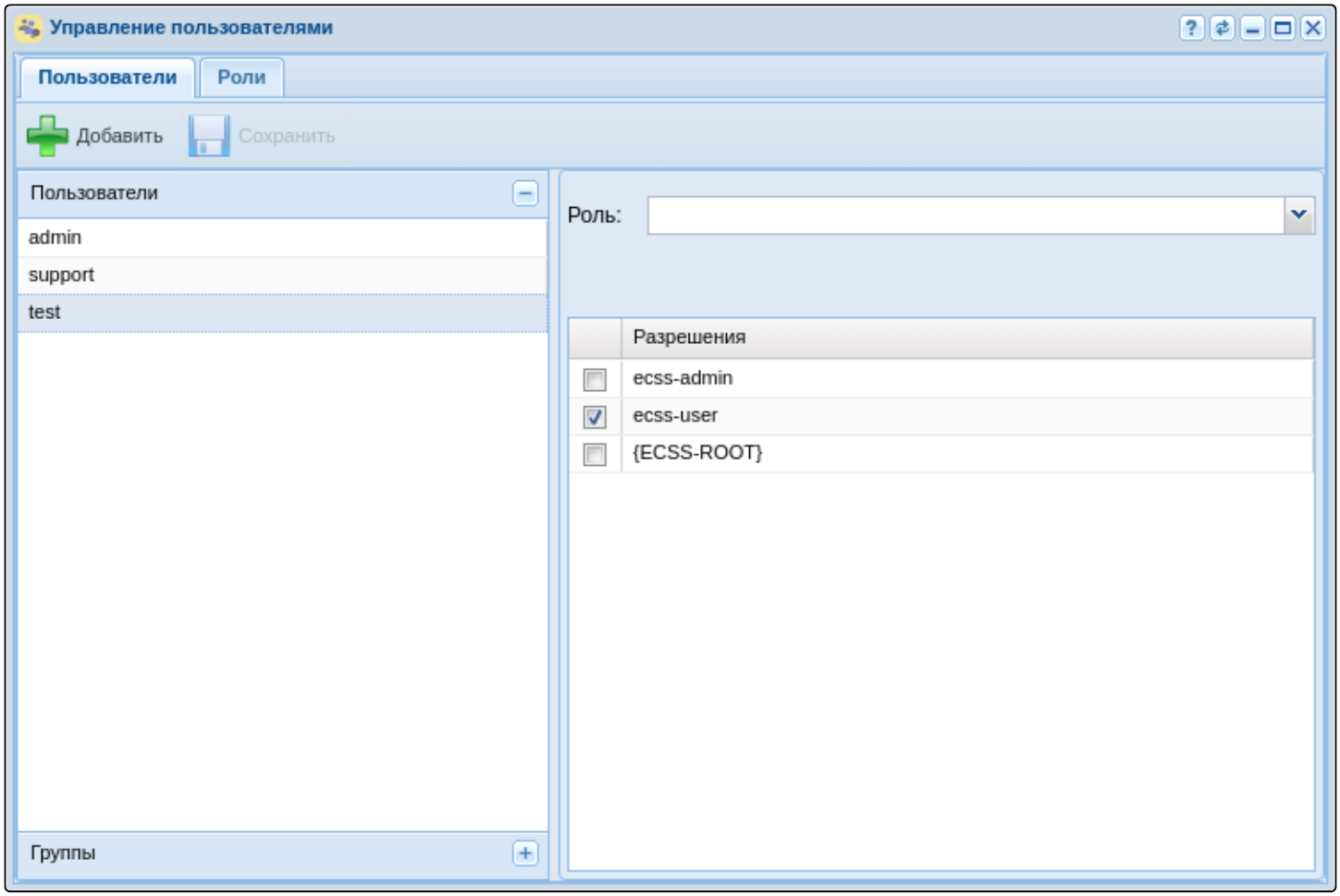


Рисунок 7 — Вид приложения с созданной учетной записью оператора

Для смены пароля рядом с названием пользователя нажмите на кнопку редактировать. В появившемся диалоговом окне введите:

1. Старый пароль (для пользователя "**admin**" по умолчанию паролем является **"password"**);

2. Новый пароль;

3. Подтвердите новый пароль.

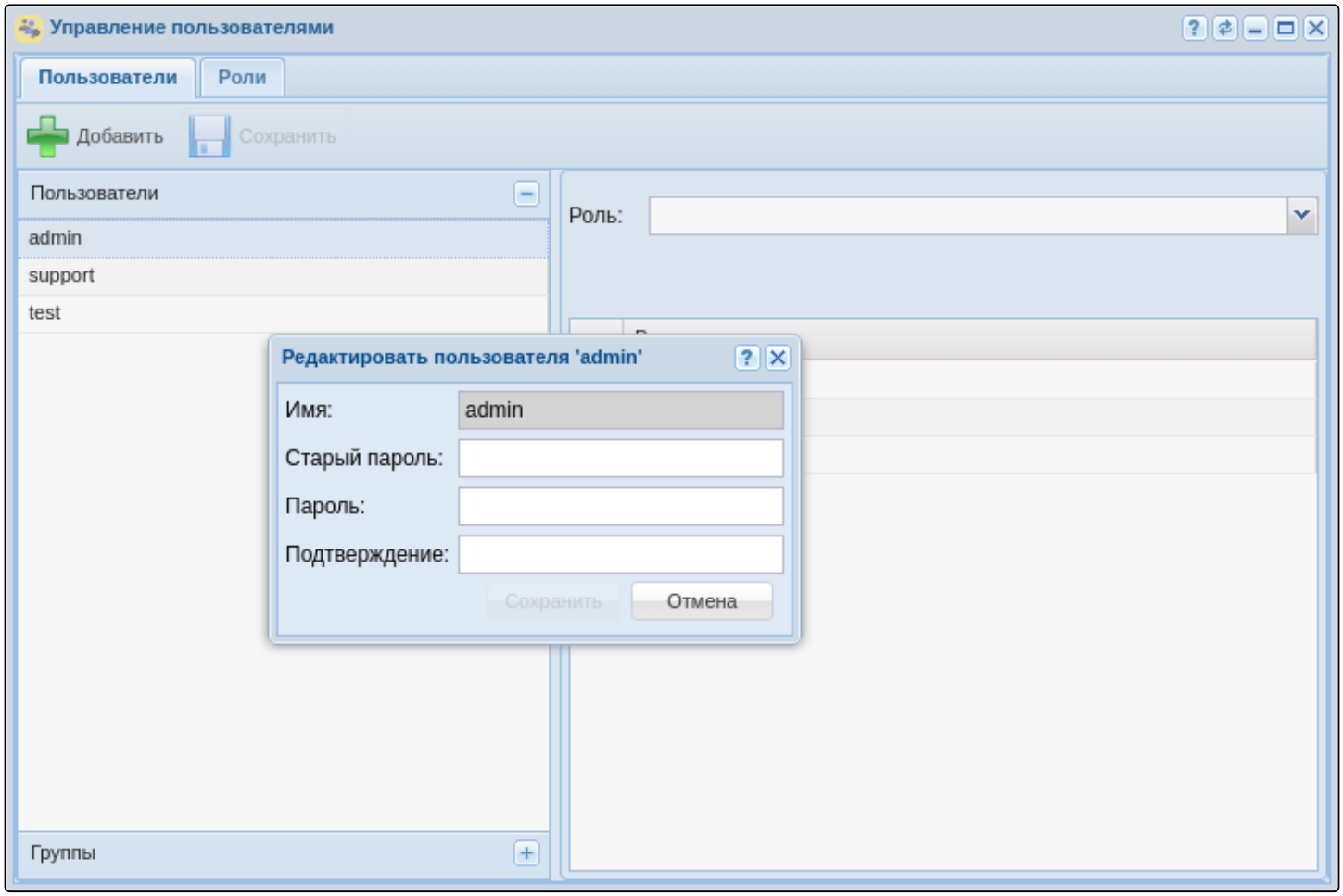


Рисунок 8 — Диалоговое окно редактирования учетной записи

14.4 Создание домена

Для создания домена необходимо войти в приложение *"Домены" ("Domain")*. В открывшемся окне требуется создать домен, для этого:

1. Нажмите кнопку "*Добавить домен*":

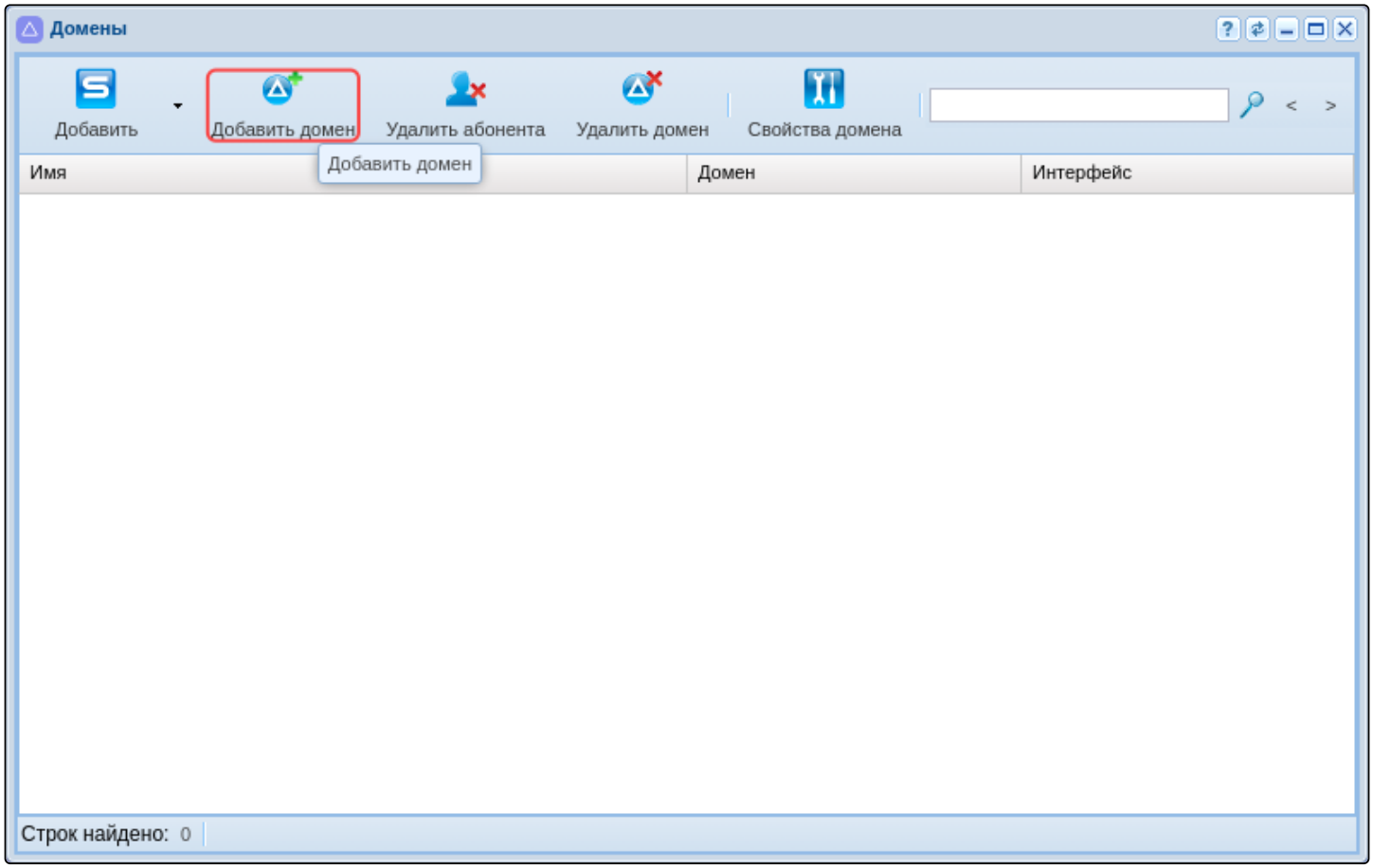


Рисунок 9 — Добавление домена в систему

2. В появившемся диалоговом окне доступны следующие настройки:

- *Имя (Name)* — индивидуальное имя виртуальной АТС;
- *Профиль услуг (SS profile)* — системный профиль дополнительных услуг. Этот профиль будет скопирован с тем же именем во вновь создаваемый домен и для всех услуг из этого профиля будет автоматически разрешён доступ через access-list;
- *IVR профиль (IVR profile)* — профиль IVR, заданный в приложении "Редактор IVR ограничений".

Введите название домена, например "test_domain";

3. Нажмите кнопку "*Ок*":

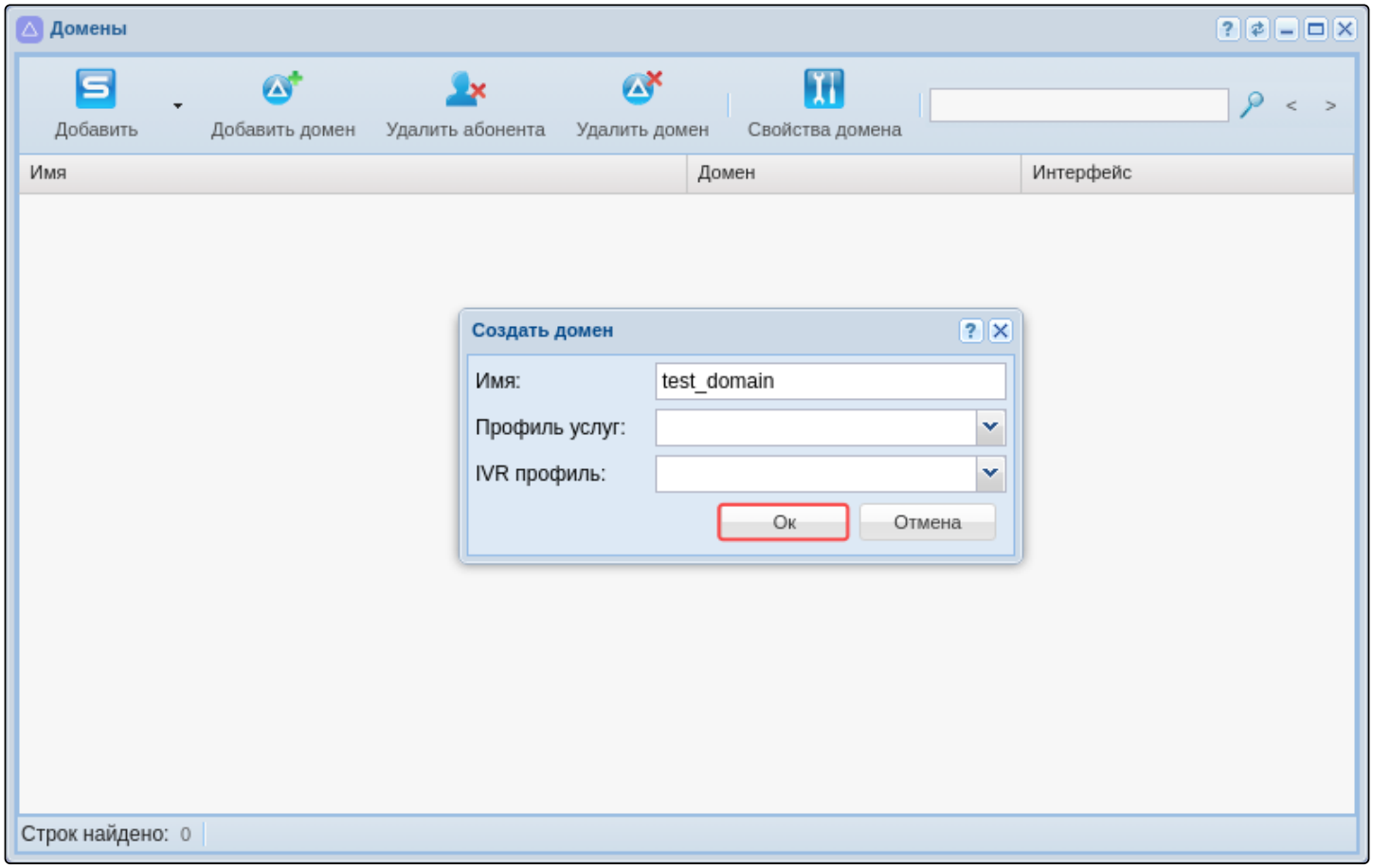


Рисунок 10 – Определение настроек нового домена

4. Нажмите кнопку *"Обновить"* в приложении.

Созданный домен будет отображаться в текущей конфигурации:

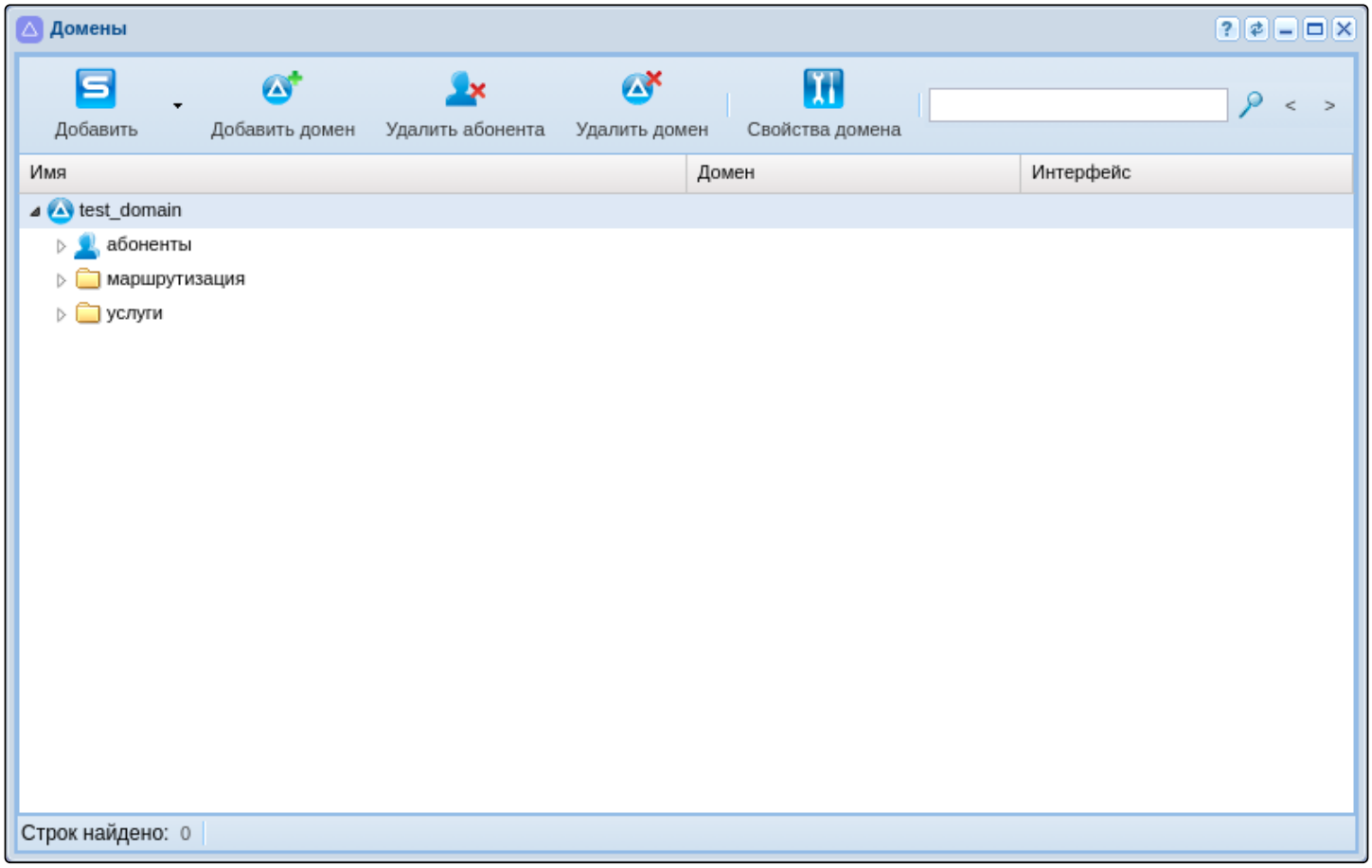


Рисунок 11 — Отображение созданного домена

Для редактирования текущего домена его необходимо выбрать в системе. Для перехода в домен воспользуйтесь опцией выбора домена (см. пункт 2 на рисунке "Вид рабочей области web-конфигуратора").

После выбора домена, согласно текущей конфигурации системы, станут доступны все приложения:

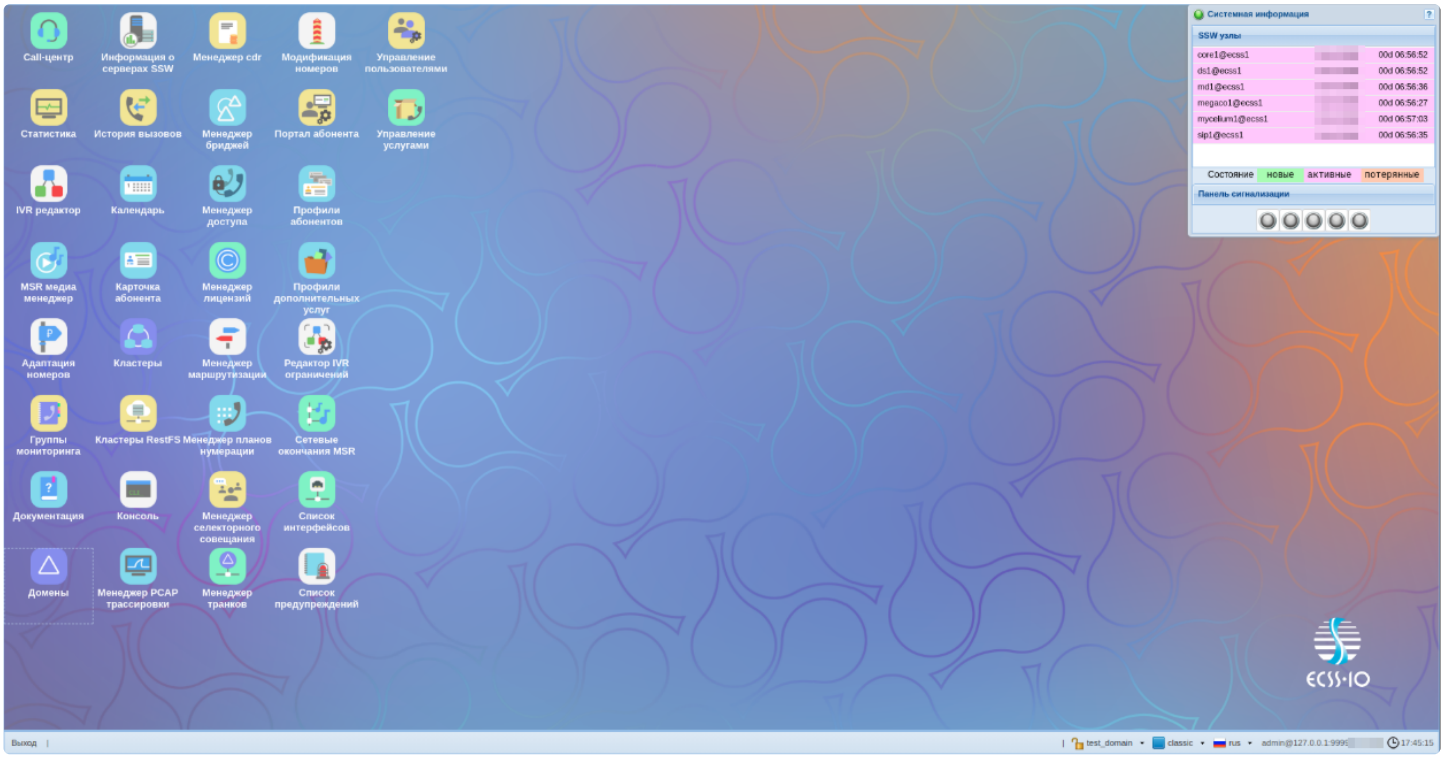


Рисунок 12 — Отображение приложений в текущей конфигурации системы

14.5 Создание IP-set (sip-транспорта) и присвоение его к домену

Для настройки интерфейса необходимо открыть приложение "Кластеры ("Clusters").

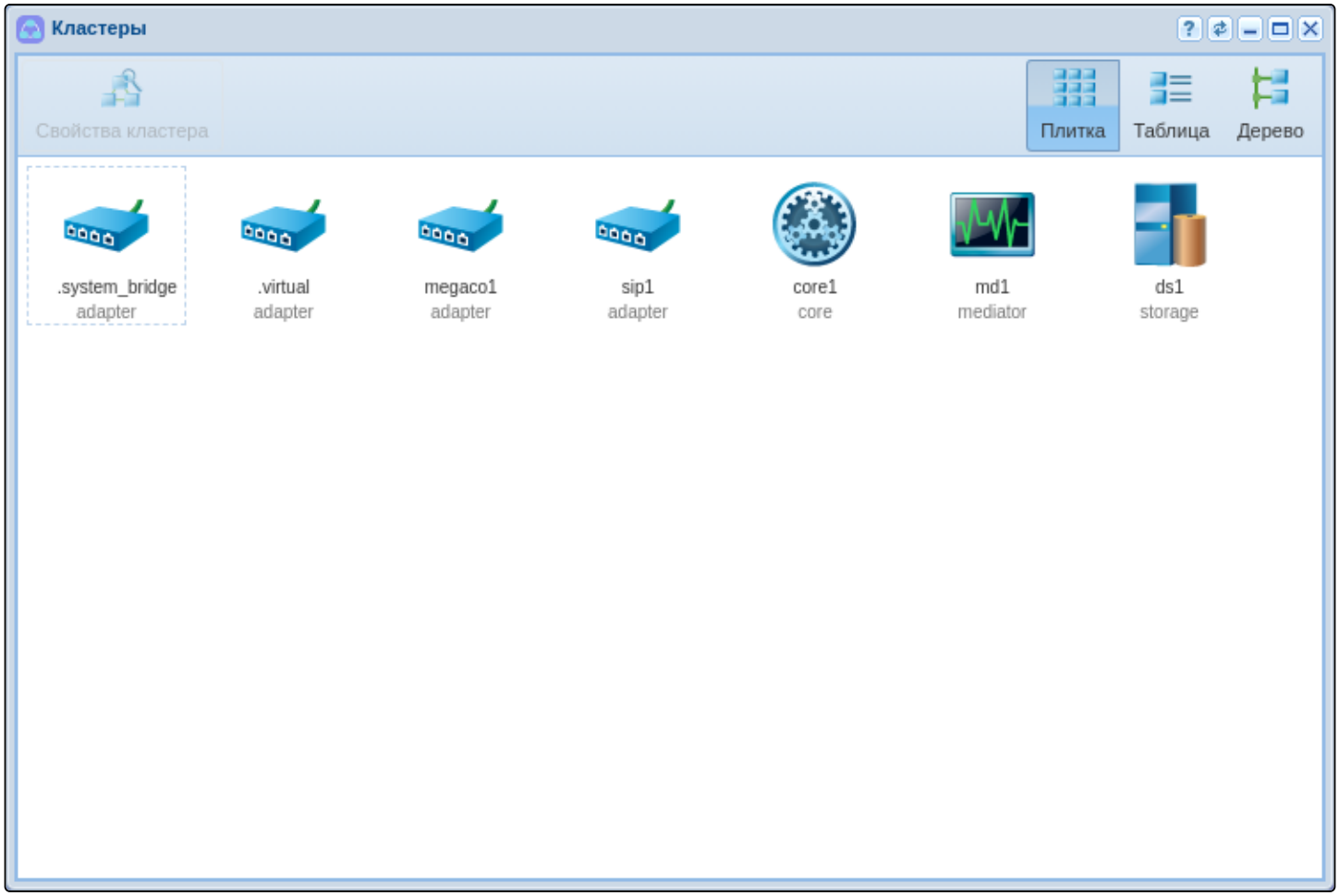


Рисунок 13 — Вид приложения "Кластеры"

IP-set — совокупность одного или нескольких IP-адресов из одной подсети и списка слушающих UDP/TCP-портов, которые будут открыты SIP-адаптером на данных адресах.

Для создания новой группы IP-адресов (IP-set) выделите кластер SIP-адаптера "sip1" и нажмите на кнопку *"Свойства кластера"* (или нажмите на иконку кластера двойным щелчком левой кнопки мыши).

В появившемся диалоговом окне перейдите на вкладку *"Транспорт"*. Далее нажмите кнопку *"Добавить"*. Появится новая группа, для редактирования полей двойным щелчком кнопки мыши нажмите на нужное:

1. Переименуйте группу адресов (IP-set), например "test_set";

2. Задайте порт, по которому будет происходить обращение к домену, например 5062;

3. Разверните вновь созданную группу нажатием на треугольник слева от названия группы;

4. Определите адрес для ноды SIP-адаптера, согласно примеру конфигурации. Для системы без резервирования укажите 10.0.20.10:

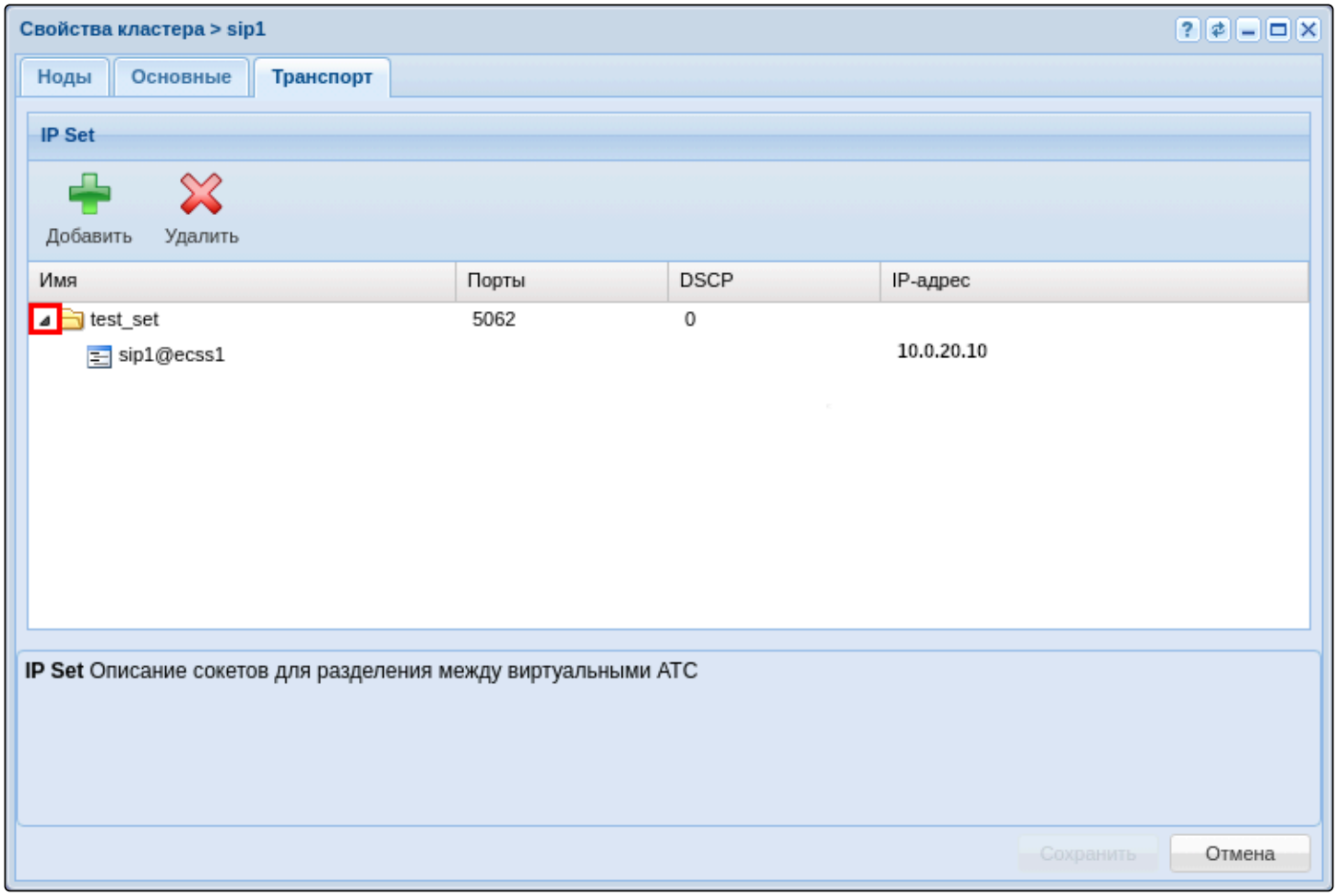


Рисунок 14 — Вид приложения "Кластеры/Транспорт"

Для системы с резервированием укажите 10.0.20.31 и 10.0.20.32:

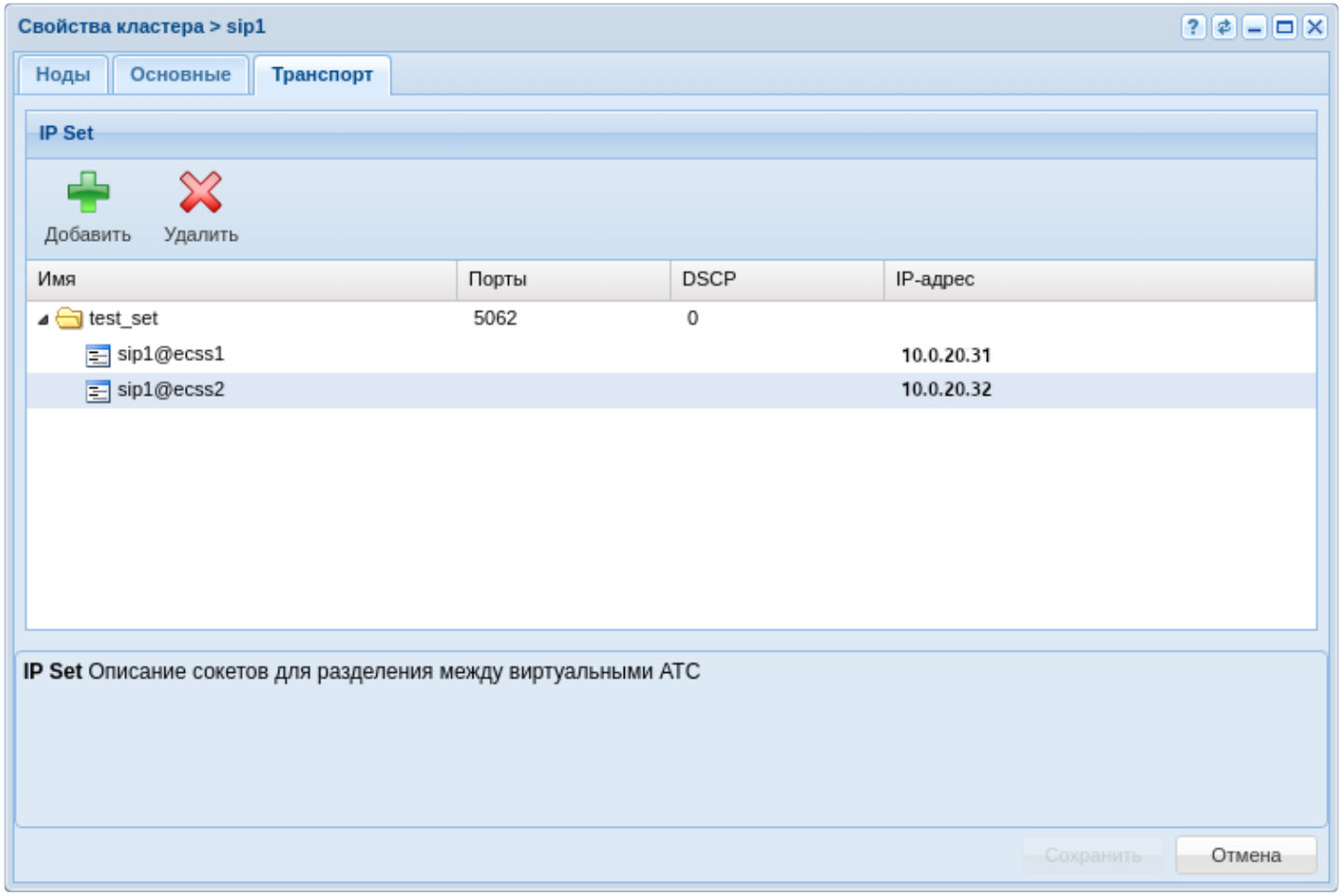


Рисунок 15 — Вид приложения "Кластеры/Транспорт" для кластера

Нажмите кнопку *"Сохранить"* для применения настроек.

Для того чтобы привязать группу адресов к домену, необходимо вернуться в приложение *"Домены" ("Domain")*, выделить домен и перейти в настройки нажатием на кнопку *"Свойства домена"* или двойным нажатием левой кнопки мыши на домен.

В настройках списка откройте ветку "SIP", в ней "SIP транспорт", далее выберите в поле IP set созданную группу адресов. Нажмите кнопку *"Сохранить"* для применения настроек.

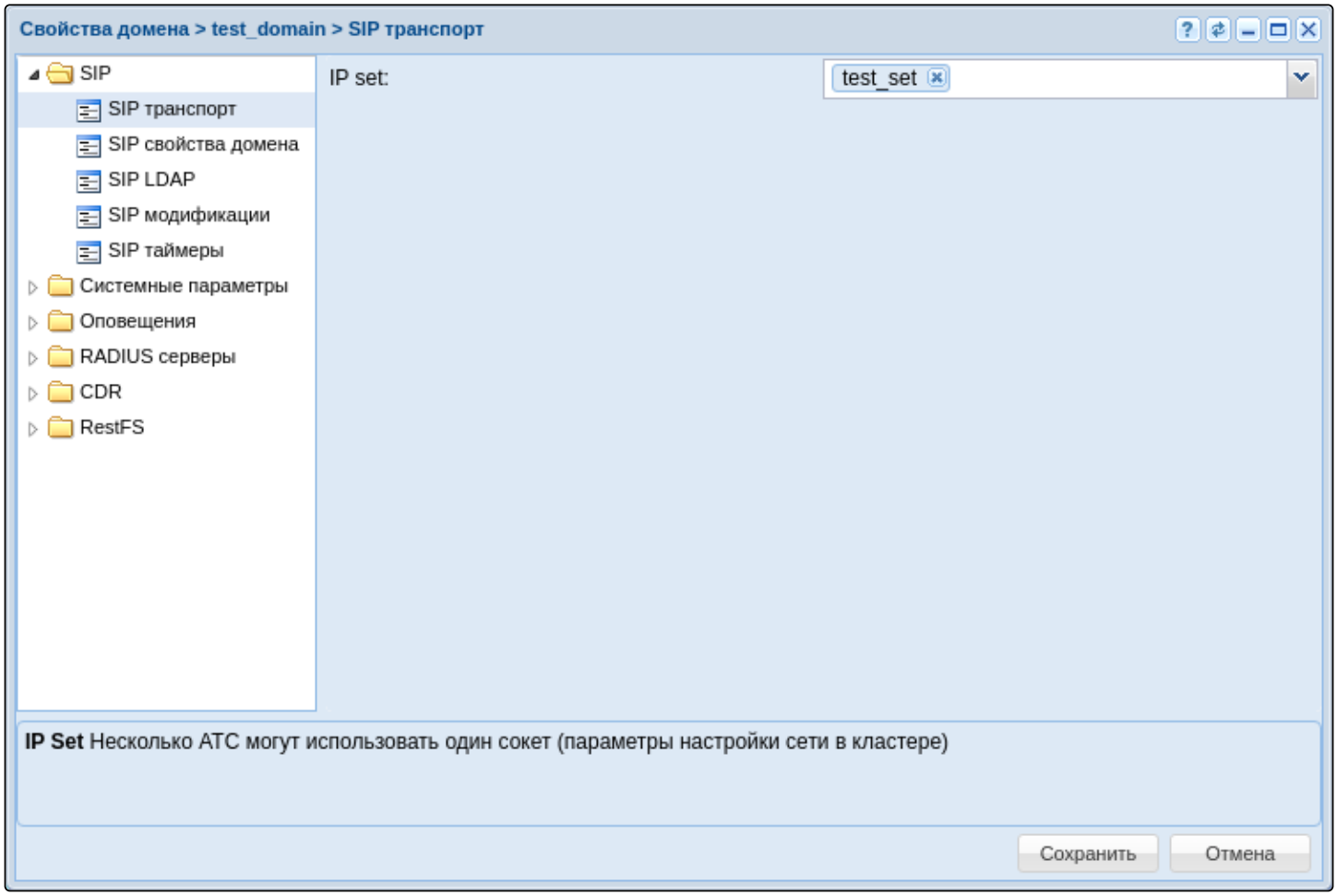


Рисунок 16 — Окно настройки для SIP транспорта

14.6 Создание абонентов

Для создания и редактирования параметров абонентов в системе используется приложение "Карточка абонента" ("Subscriber card").

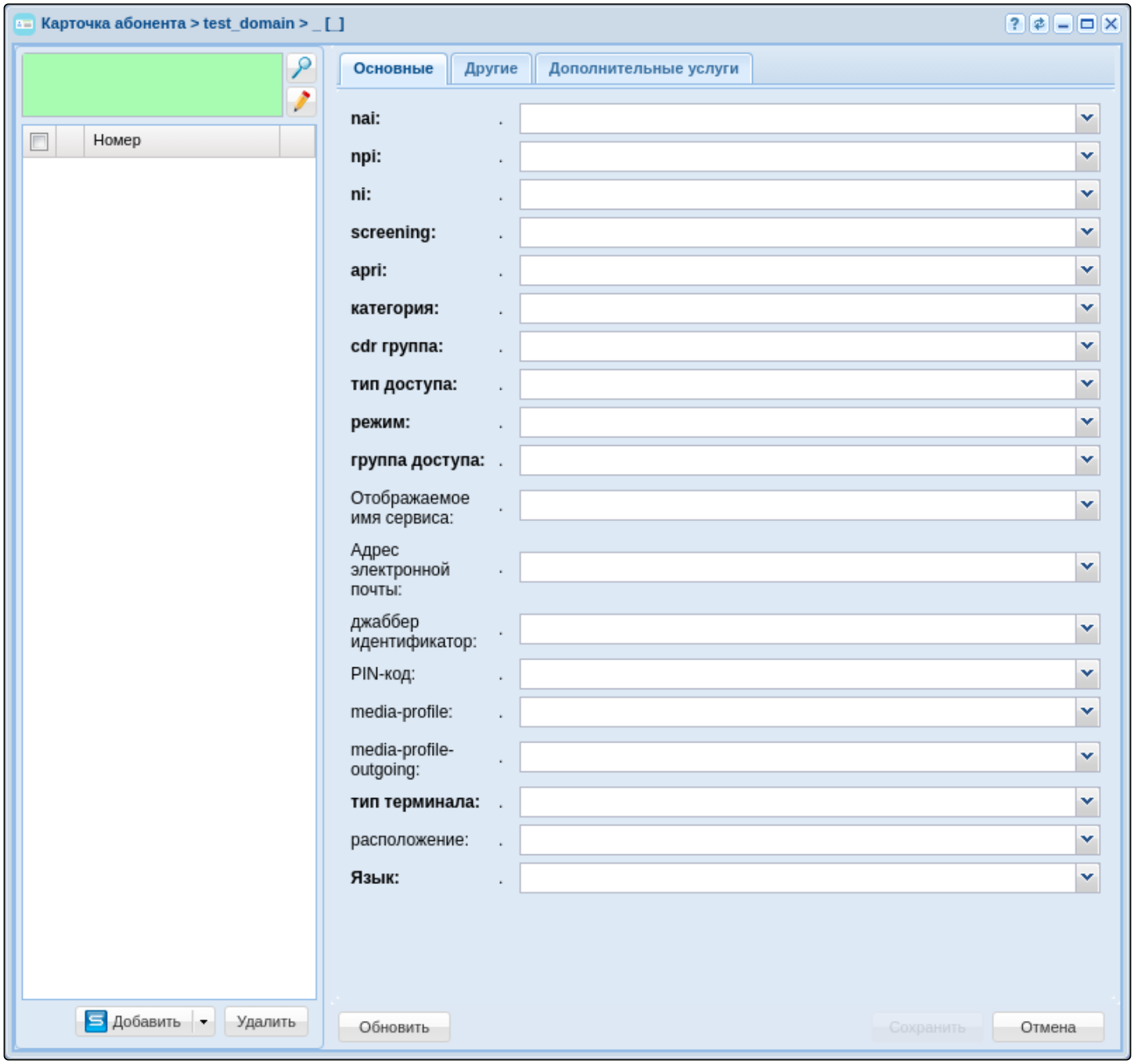


Рисунок 17 — Вид приложения "Карточка абонента"

В конфигурации возможно создание SIP-абонентов и виртуальных абонентов.

Для пользователей, имеющих физическое окончание, используется функционал SIP-абонента, в то время как виртуальный абонент используется тогда, когда нужна функциональность без физических окончаний. Например, номер для обращения к ivr-скрипту.

Для создания новых абонентов нужно нажать на кнопку *"Добавить"*.

В появившемся диалоговом окне укажите следующие параметры:

- *Контекст* — контекст маршрутизации, выбираем тот, который создали, например "test_name";
- *Имя интерфейса* — номер или группа номеров, который присваивается абоненту, например {100-110};
- *Алиас как пользователь* — настройка, которая связывает номер сущности, алиас и пользователя одним и тем же названием, в примере настройка активирована;

- *Авторизация* – порядок проверки подлинности прав пользователя на доступ к данным, в примере используется always;
- *Логин* – использую настройку номер как логин для простоты;
- *Пароль* – можно задать или использовать сгенерированный системой.

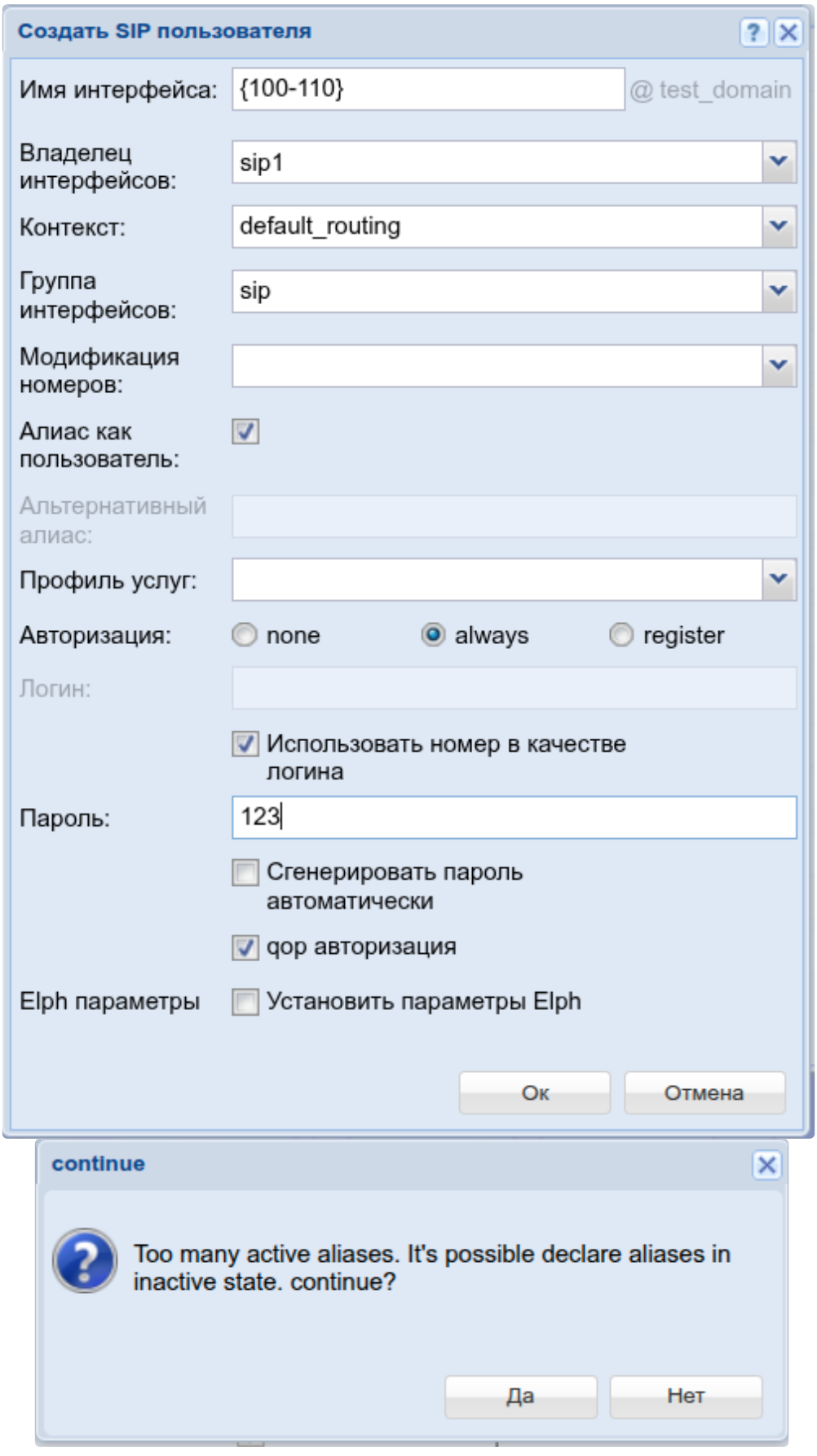


Рисунок 18 — Пример определения абонентов в домене

14.7 Создание и применение контекстов маршрутизации для домена

Маршрутизация отвечает за поиск номера и последующую адресацию вызова. Для корректной работы системы должен быть настроен как минимум один контекст маршрутизации.

Настройка маршрутизации осуществляется в приложении "Менеджер маршрутизации" ("Routing manager") .

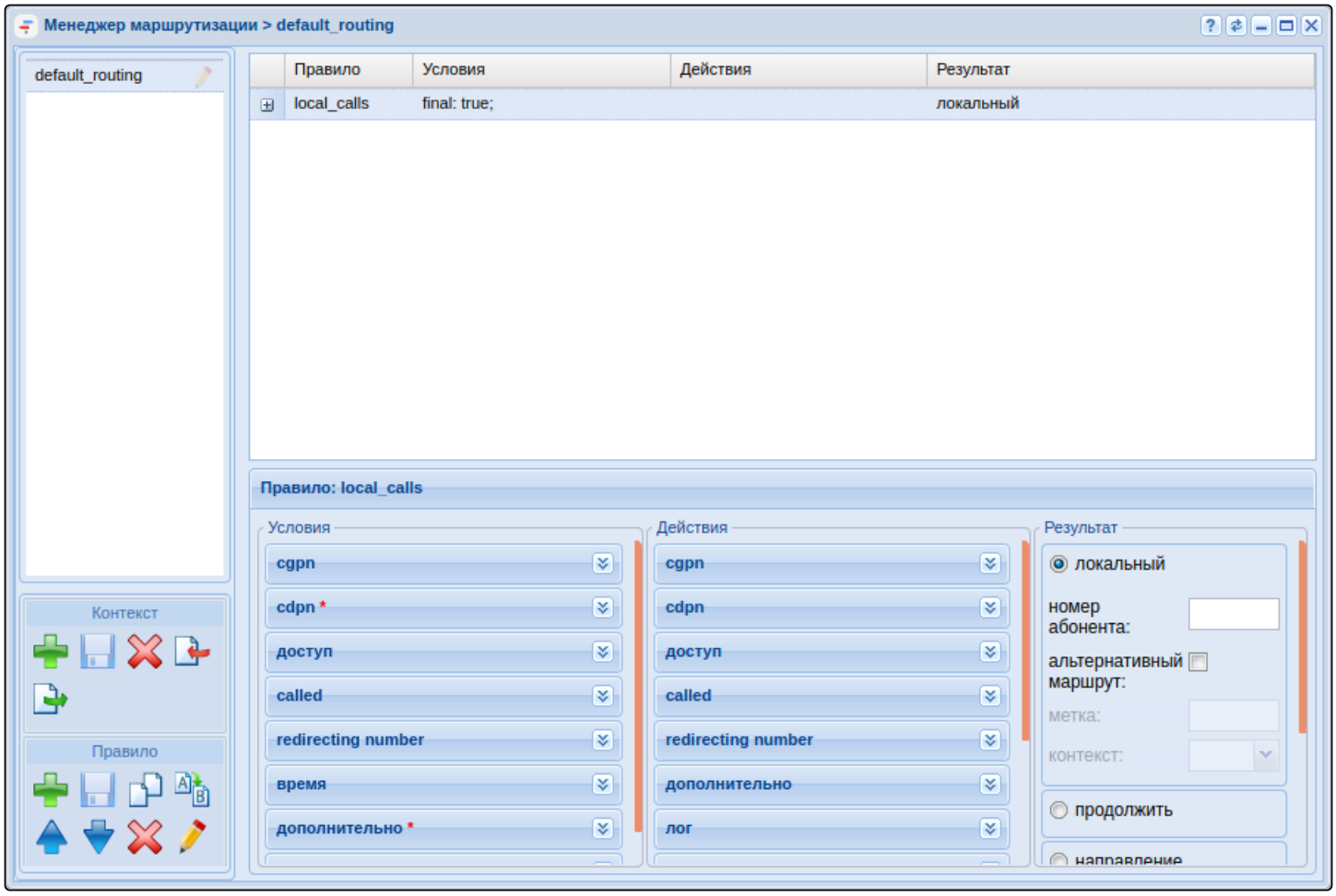
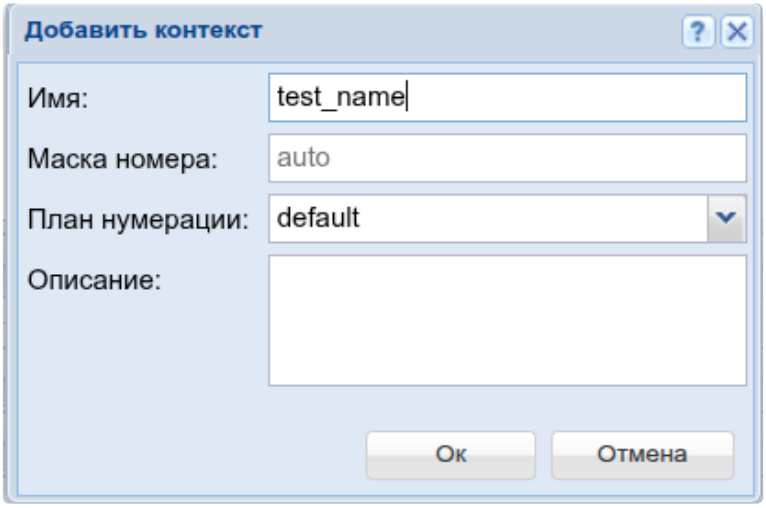


Рисунок 19 — Вид приложения "Менеджер маршрутизации"

В качестве примера создадим контекст, а затем несколько правил в нём, для этого:

1. В левой части окна в разделе "Контекст" нажмите на кнопку *"Создать контекст"* ;
2. В открывшемся диалоговом окне обозначьте имя контекста, а также тип контекста — пустой контекст:



3. Нажмите *"Сохранить контекст"* ;

Создадим 4 правила в данном контексте:

- rule1 — правило для выхода на транк TAU-72;
- rule2 — правило локальной маршрутизации для номеров 101-105;
- rule3 — правило для выхода в ivr;
- rule4 — правило-исключение.

Для создания нового правила нужно выделить созданный контекст и нажать кнопку *"Создать правило"*. В появившемся окне нужно ввести название правила. Затем сохранить вновь созданные правила.

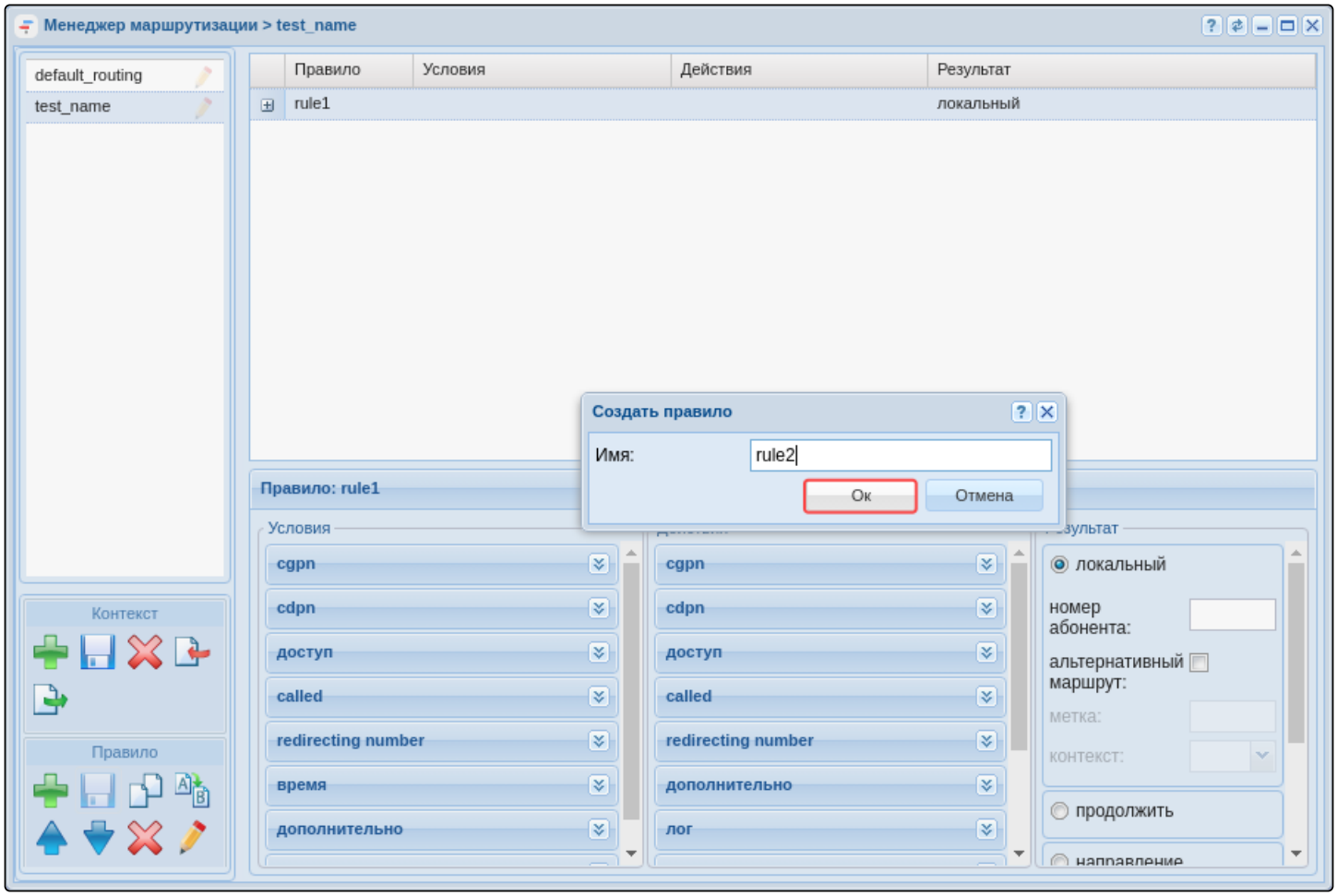


Рисунок 20 — Создание правила контекста маршрутизации

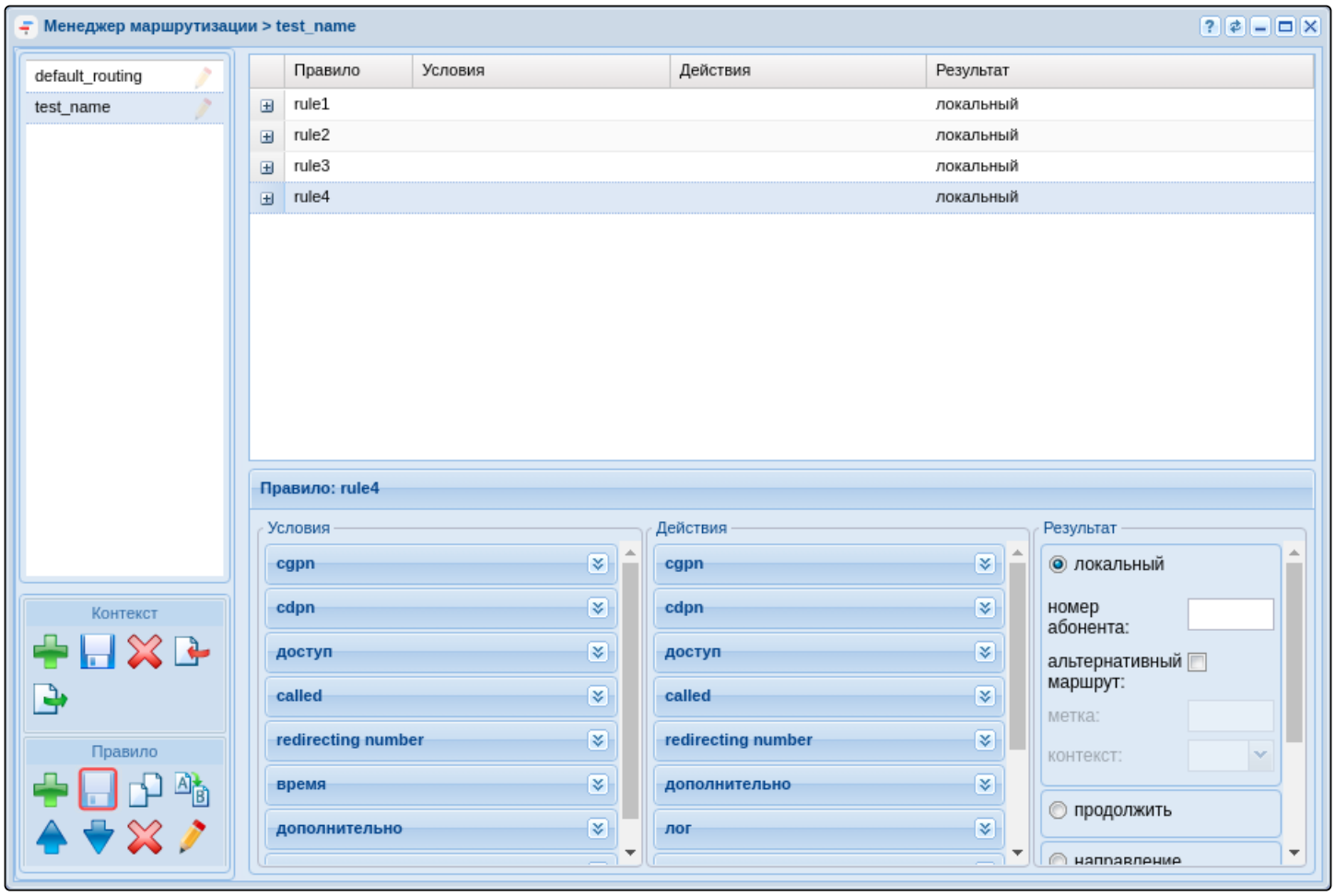


Рисунок 21 — Определение правил

На данный момент не определен транк, чтобы можно было сослаться на него в правиле, однако можно задать номера, по которым будет производится выборка.

Нажав на правило rule1, перейдем в нижнюю часть экрана, в которой располагаются области для редактирования контекста маршрутизации. Условно определим, что выборка для выхода на транк будет проводиться по признакам номера вызываемого абонента (CDPN), а номера в транке должны начинаться с цифры 4.

Функционально контекст маршрутизации делится на три части:

- Условия ("Condition") — раздел, определяющий выражения для выборки по предложенным признакам;
- Действия ("Action") — раздел, выполняющий преобразование признаков номеров к конкретному значению;
- Результат ("Result") — раздел, завершающий маршрутизацию и определяющий её результат.

rule1: Для выхода на транк нужно правильно отредактировать каждую часть:

- В разделе условий перейти на вкладку CDPN. Ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера с 106 по 107, нужно ввести условие: 10(6-7);
- В разделе действий перейти на вкладку CDPN. Ввести маску для изменения номера в поле "Номер". Например, чтобы добавить число 4 перед номером, нужно заполнить поле следующим выражением: 4{1,2,3};
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить транк в системе, поэтому к настройке данного правила вернемся чуть позже.

Таким же образом нужно настроить оставшиеся правила.

Для rule2:

- В разделе условий, во вкладке CDPN ввести номера телефонов, которые находятся за транком в поле "Номер". Например, чтобы определить номера со 101 по 105, ввести условие 10(1-5);
- В разделе результата определить результат как локальный (т.е. локальная маршрутизация);
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Для rule 3 допустим, что абоненты с номерами 108, 109 и 110 попадают в ivr-скрипт–информатор, прежде чем звонить далее.

- В разделе условий, во вкладке CGPN ввести номера телефонов, по которым будет выборка в поле "Номер", например 1(10,08-09).
- Нажать кнопки *"Сохранить правило"* и *"Сохранить контекст"* для применения изменений.

Чтобы настроить поле результата, нужно определить в системе ivr-скрипт, поэтому к настройке данного правила вернемся чуть позже.

Для rule 4 мы определяем правило-исключение — это правило, которое работает в случае, если любые другие правила не срабатывают.

- По умолчанию данное правило создается в системе в контексте default_routing. В условиях на вызываемый номер прописывается символ %, а результатом является локальная маршрутизация, однако, если мы назначаем абоненту контекст, отличный от default_routing, в конце *рекомендуется* создавать это правило.

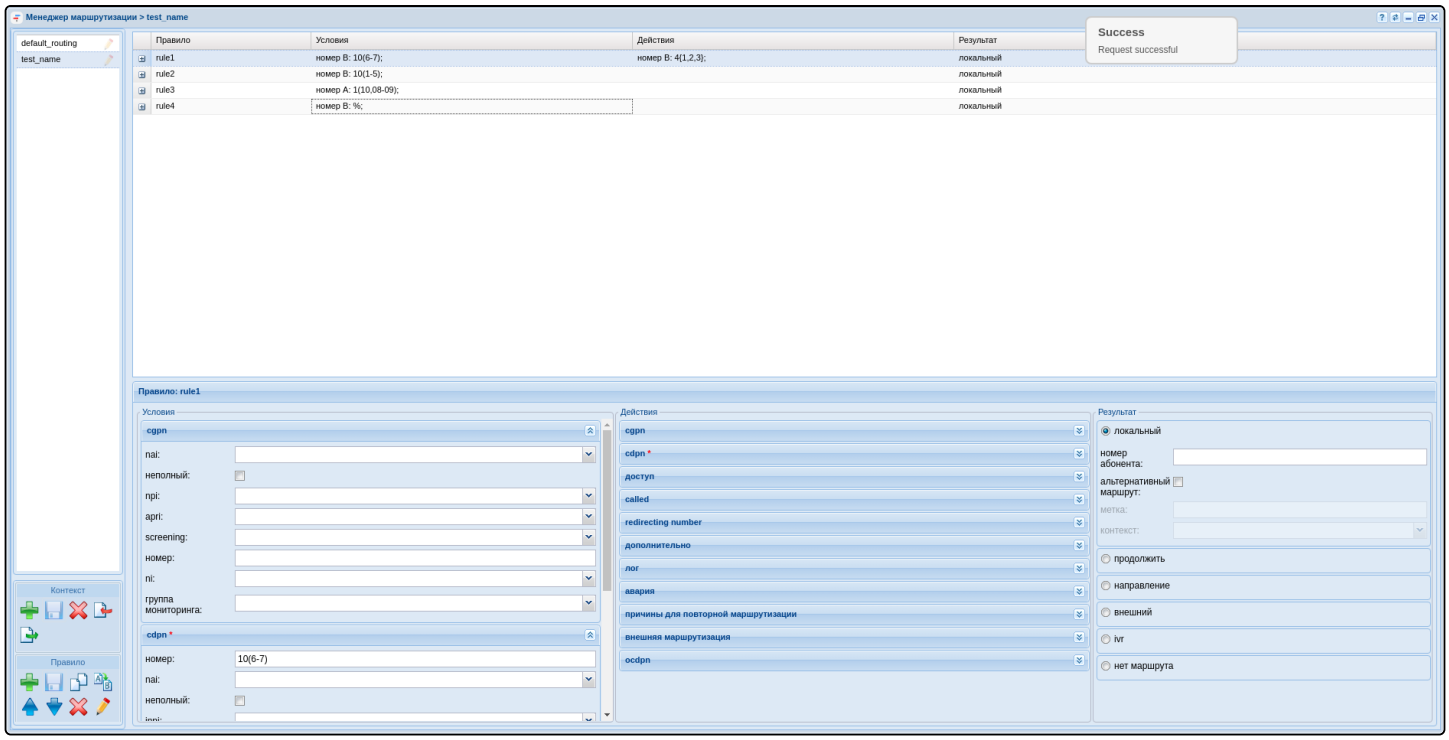


Рисунок 22 — Пример настроек контекста маршрутизации

14.8 Создание транка

Для создания и редактирования параметров транка в системе используется приложение "Менеджер транков" ("Trunk manager") .

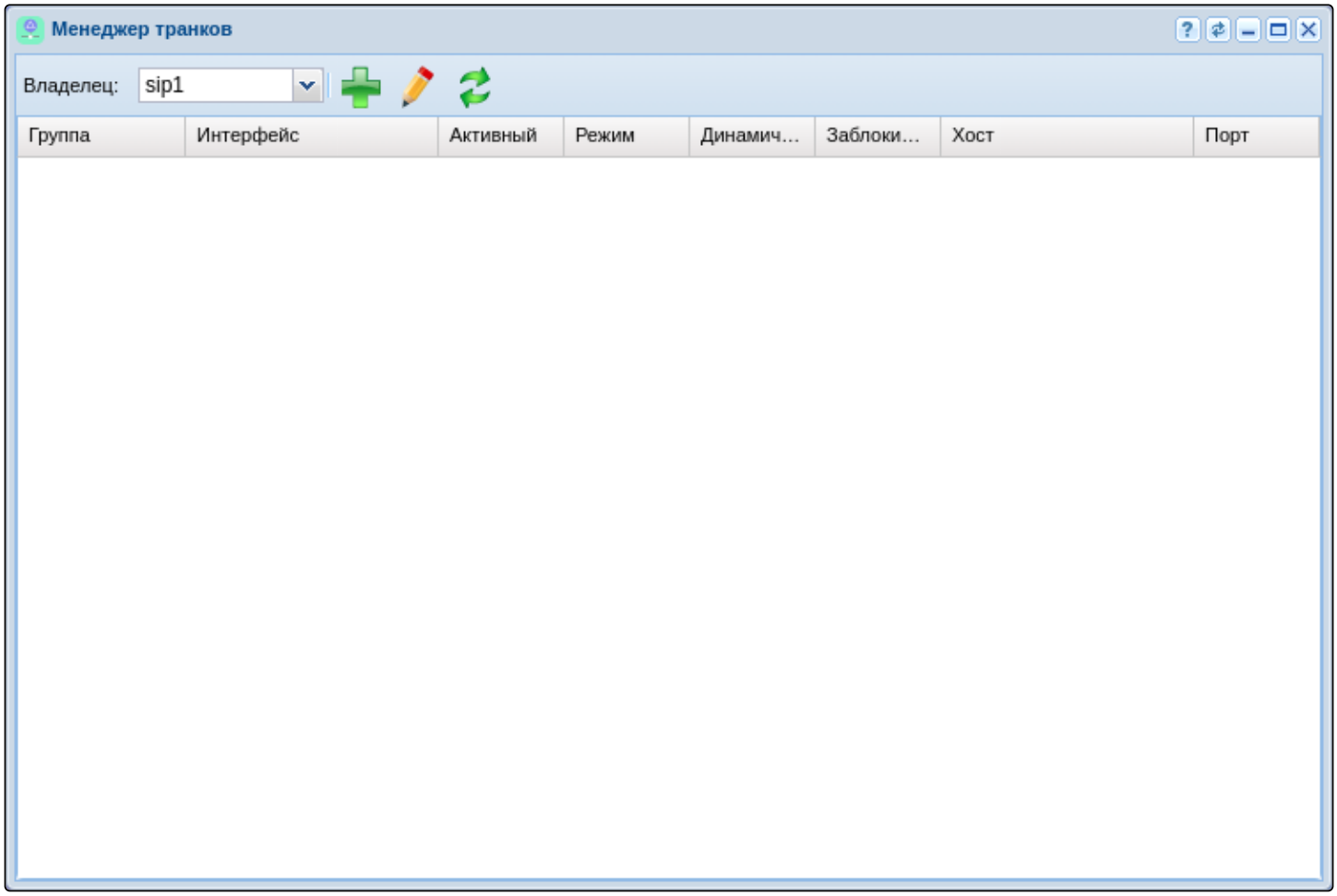


Рисунок 23 — Вид приложения "Менеджер транков"

Для определения транка в системе нужно нажать на кнопку *"Создать транк"*  , в появившемся диалоговом окне определить параметры:

- *Имя* — назначить название транка, по которому его можно идентифицировать в системе;
- *Контекст* — применить ранее созданный контекст маршрутизации test_name;
- *Группа* — выбрать группу интерфейсов, созданную при определении абонентов "test.group";
- *Группа IP-адресов (IP-set)* — созданная на домене группа адресов "test_set";
- *Регистрация* — если используется транковая, то нужно включить данный параметр, в примере она не используется;
- *Хост [:порт]* — IP-адрес назначения транка — 10.0.3.100;
- *Порт для приёма* — транспортный порт, на котором будет слушаться трафик с транка, соответствует порту, назначенному на группу IP-адресов.

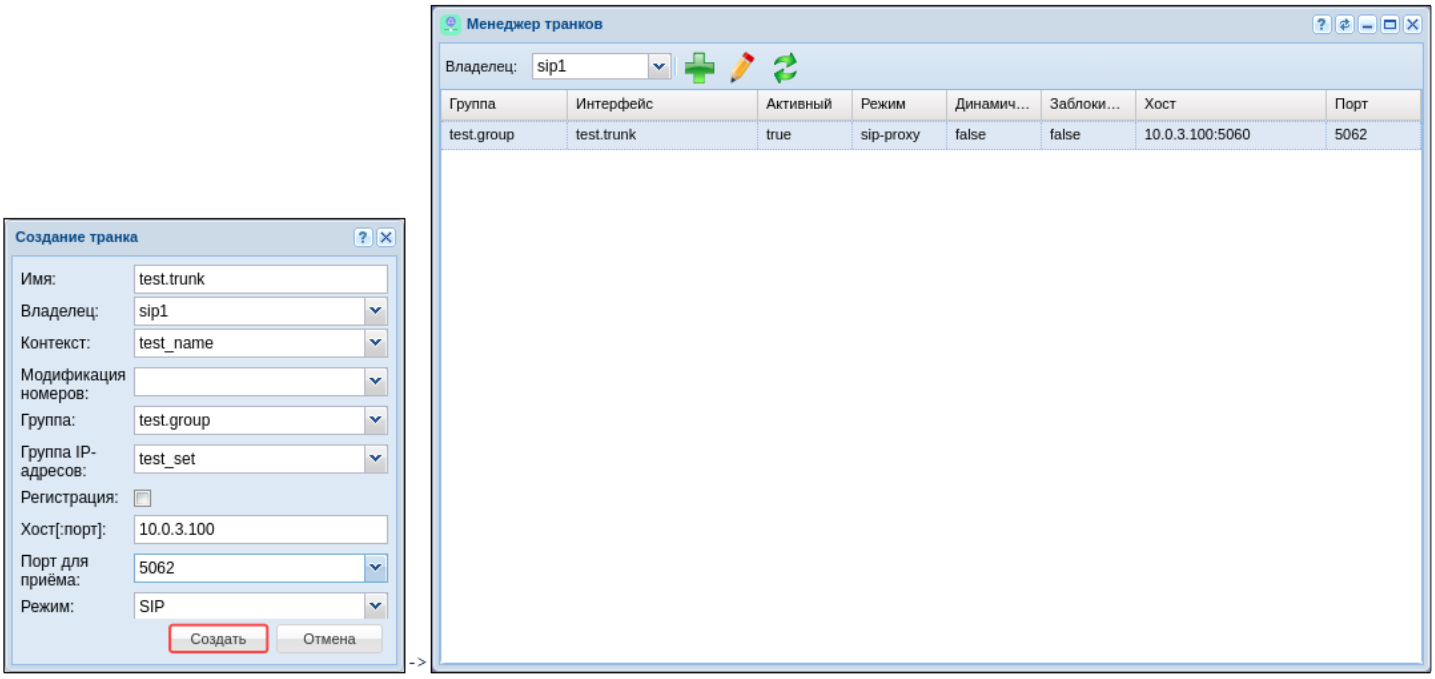


Рисунок 24 — Создание транка

14.9 Создание IVR-скрипта

Для создания IVR используется приложение "IVR-редактор" ("IVR editor").

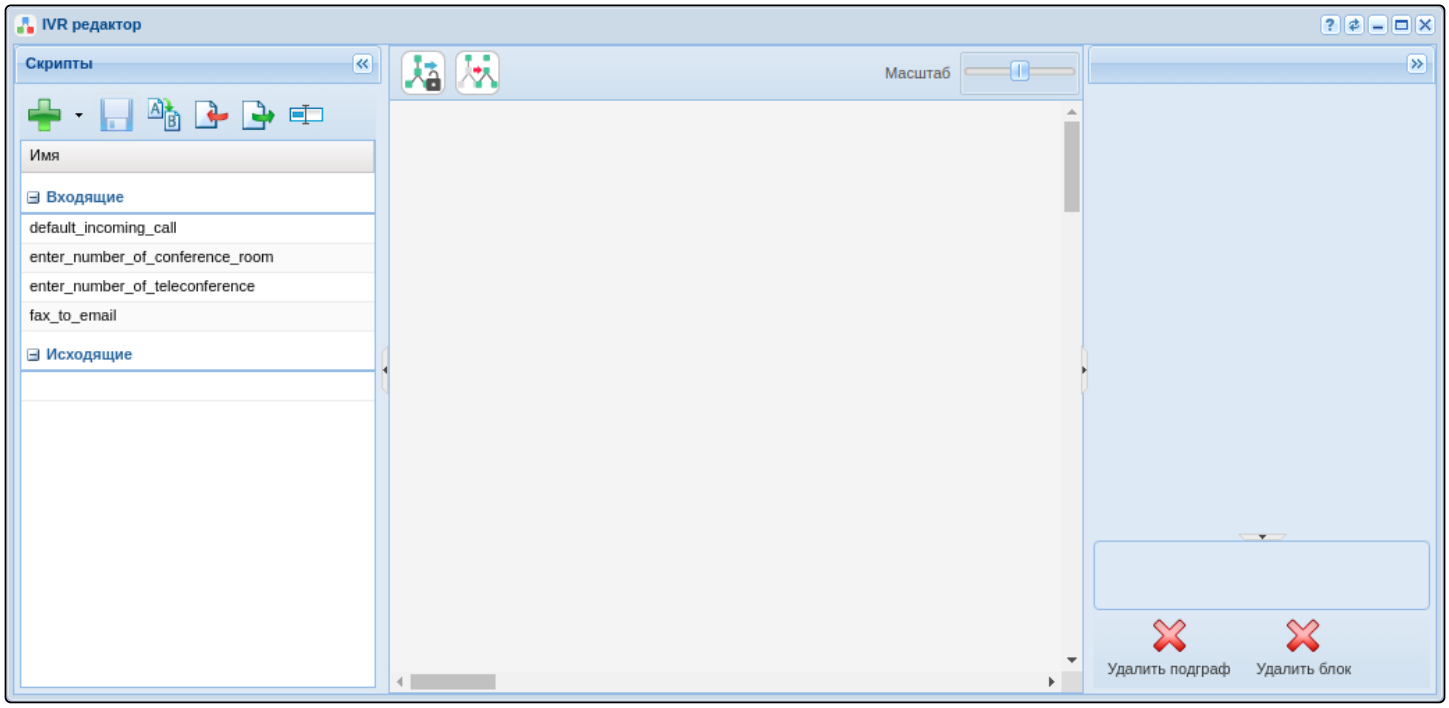


Рисунок 25 — Вид приложения "IVR-редактор"

Для создания скрипта нужно нажать на кнопку *"Добавить"* , выбрать тип скрипта (в данном случае будет скрипт на входящие вызовы), в диалоговом окне указать название скрипта, например "test_ivr".

После создания скрипта в основном окне редактора появится блок-схема.

В качестве примера составим скрипт, который при срабатывании проигрывает вызывающему абоненту заранее записанную фразу, а затем продолжает вызов.

Пример скрипта приведен на рисунках ниже. Здесь используются блоки Info и Dial.

- Блок Info — блок, который проигрывает сообщения до момента ответа пользователя. В качестве записанной фразы используется генератор тонов:

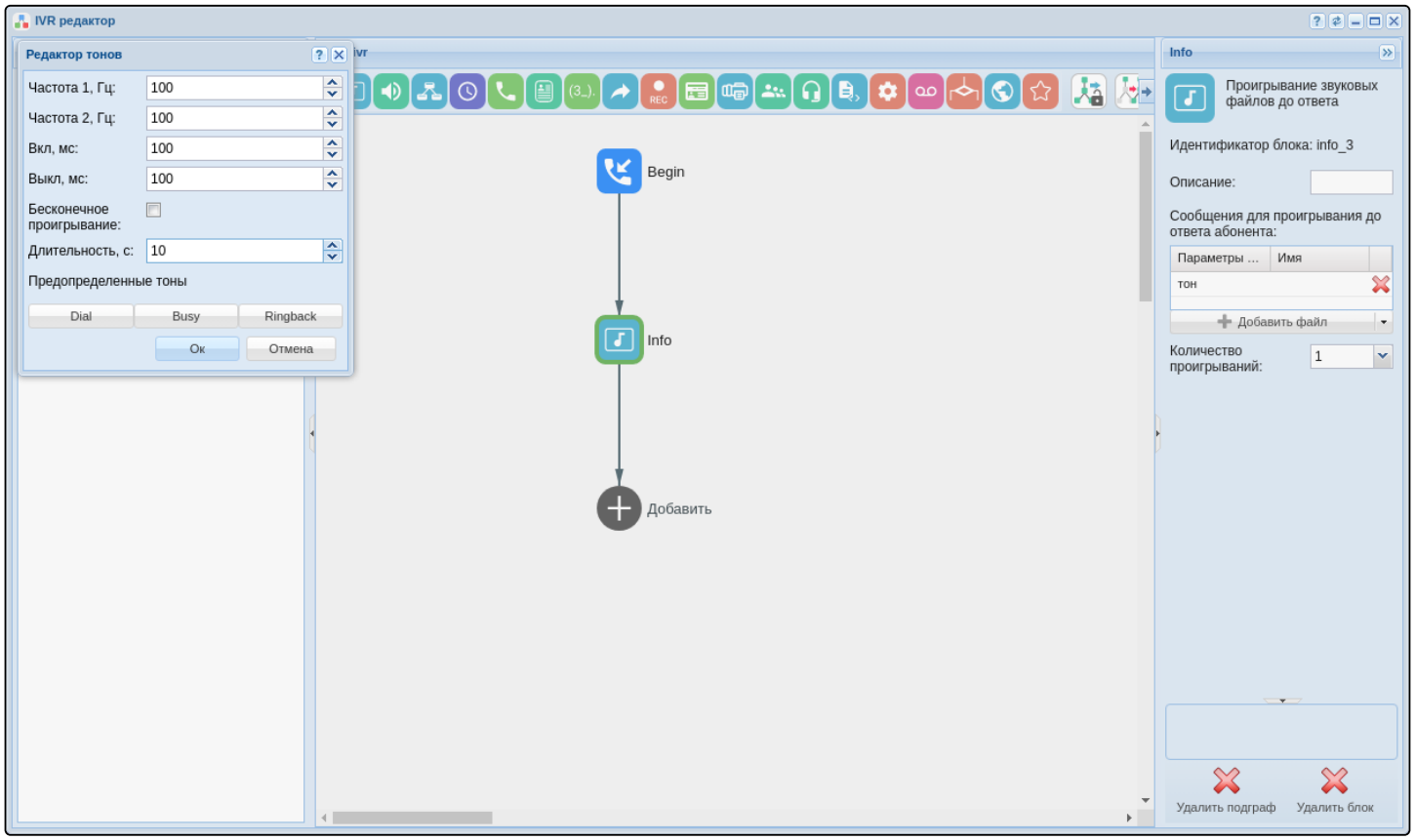


Рисунок 26 — Вид рабочей области IVR с настройками блока "Info"

- Блок dial — блок, который совершает вызов на заданный номер. Чтобы продолжить вызов, нужно применить предопределенную переменную CDPN:

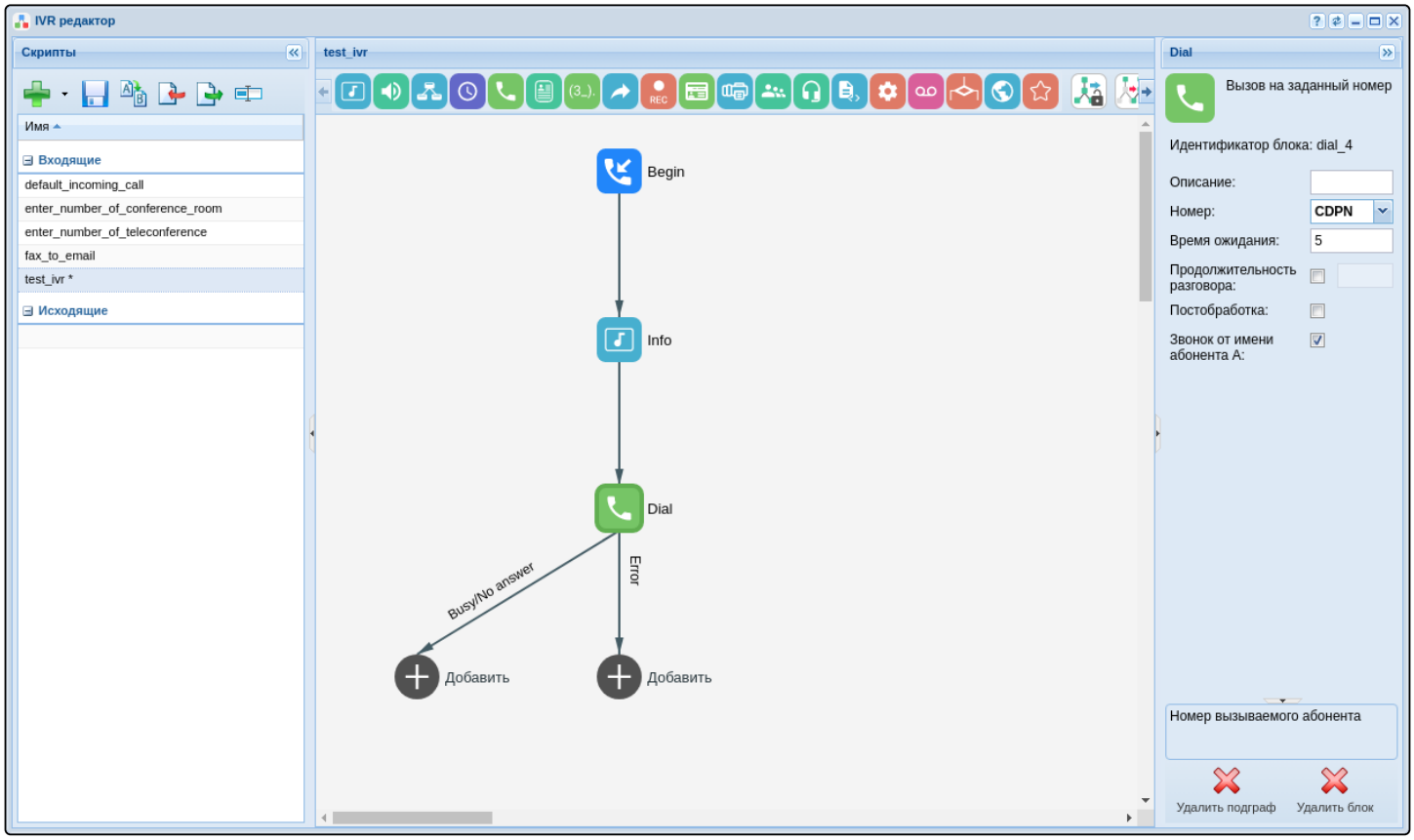


Рисунок 27 — Вид рабочей области IVR

14.10 Завершение настройки маршрутизации

Для завершения настройки маршрутизации нужно войти в приложение "Менеджер маршрутизации" ("Routing manager") и в правилах rule1 и rule3 настроить соответствующие результаты маршрутизации, для этого:

- rule1: в разделе "Результат" выберите подпункт "внешний". В таблице "Направления" в поле "Значение" добавьте созданный ранее транк, после чего сохраните правило .
- rule3: В разделе "Результат" выберите подпункт "ivr" и в поле скрипт добавьте созданный ранее ivr-скрипт, после чего сохраните правило и контекст .

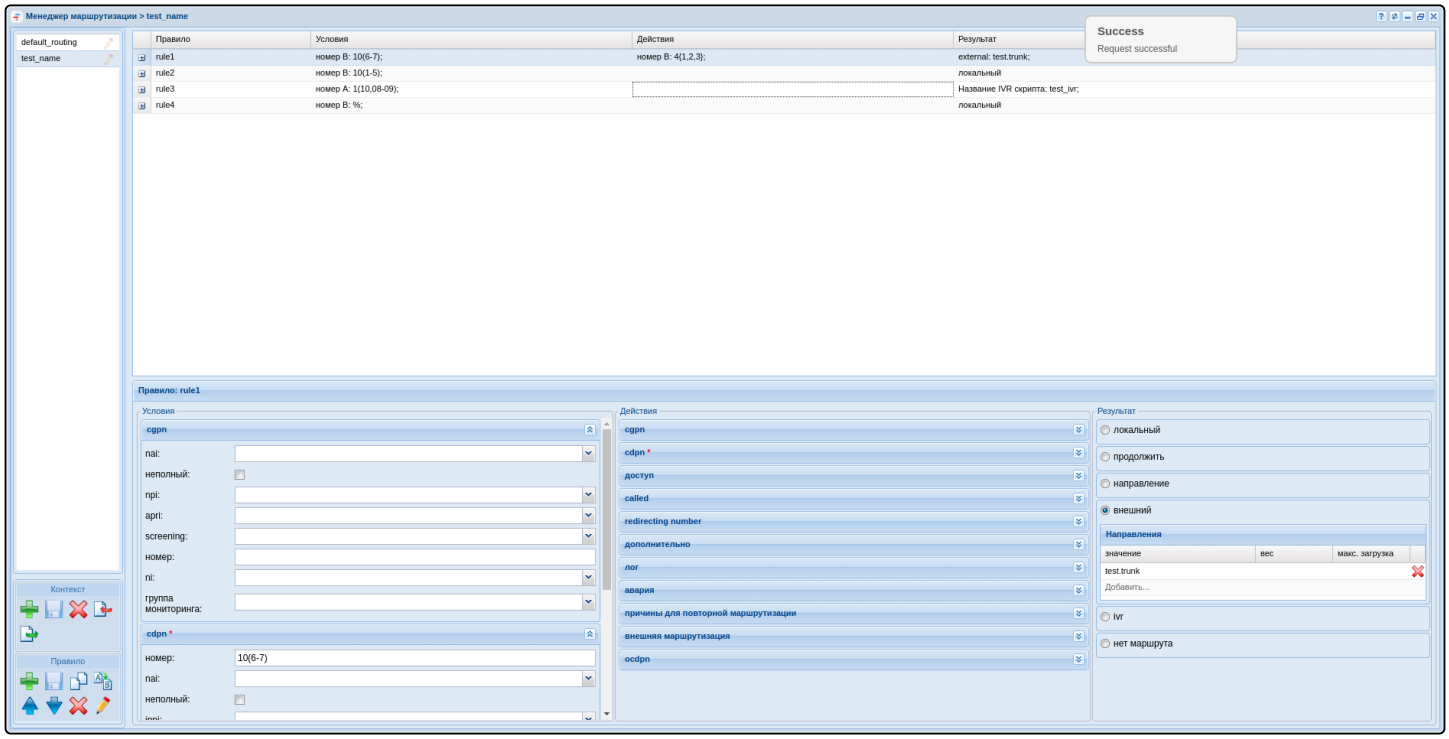


Рисунок 28 — Вид контекста маршрутизации

14.11 Настройка услуг

Для настройки услуг необходимо выполнить несколько действий:

- Установить услуги через CoCon;
- Добавить услуги для домена в список доступа (access-list) через CoCon;
- Далее любая услуга в списке доступа становится применимой на абоненте или транке.

Для входа в CLI "CoCon" можно воспользоваться терминалом или приложением "Командная консоль CoCon" ("CoCon").

После авторизации в Cocon пропишите следующие команды:

- Для установки услуг в систему:

```
cluster/storage/ds1/ss/install ds1@ecss1 *
```

- После успешной установки услуг пропишите следующую строку для добавления доступа к ним:

```
cluster/storage/ds1/ss/access-list add test_domain *
```

Символ "*****" означает, что команда будет применена ко всем доступным элементам в системе. Если требуется установить конкретную услугу, вместо "*****" вводится её название.

Для подключения услуг абоненту необходимо открыть карточку абонента, выбрать абонента из списка и перейти на вкладку "*Дополнительные услуги*".

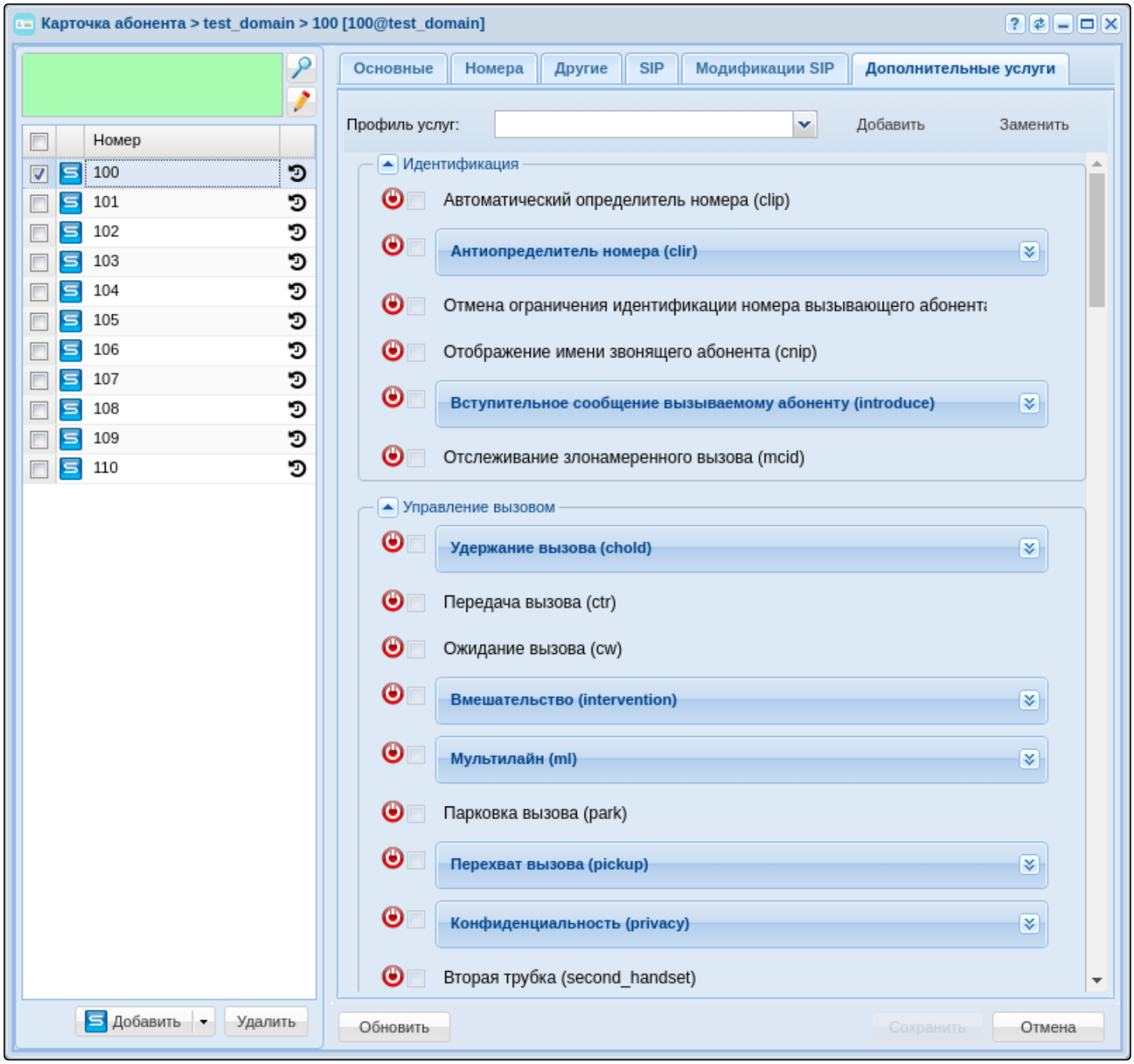


Рисунок 29 — Пример настройки услуг

Для активации услуги абоненту необходимо подключить ее, нажав кнопку , затем активировать и настроить (прочитать подробнее можно в разделе приложения "Карточка абонента" ("Subscriber card")).

14.12 Добавление медиа-ресурсов.

Для того чтобы начать работу с медиа-сервером сразу после его регистрации, необходимо создать описание известных медиа-серверов с указанием всех параметров. Приложение web-конфигуратора "Сетевые окончания MSR" ("MSR registrars") служит для настройки медиа-ресурсов.

15 Чек-лист по установке ECSS-10

Шаг	Пункт документации	Результат	Примечание
1. Проверка соответствия аппаратных требований к системе			
1.1 Установлены актуальные версии ПО внутренних устройств и контроллеров сервера (BIOS, RAID, iDRAC, iLO и т.д.)			См. документацию к используемому серверу
1.2 Наличие в системе по меньшей мере двухъядерного CPU	Разработка проекта системы		Определяется проектом на систему
1.3 Наличие в системе по меньшей мере 8 ГБ RAM	Разработка проекта системы		Определяется проектом в зависимости от требуемой нагрузки. В тестовых целях допускается уменьшение до 4 ГБ
1.4 Имеется достаточно дискового пространства	Установка ОС Ubuntu Server 22.04 или Установка ОС Astra Linux 1.8		df -h
1.5 Произведена верная разметка разделов	Таблица 1 — Вариант размещения информации в файловой системе на физических носителях для серверов		df -h
1.6 В системе **НЕ используется** файл подкачки **SWAP**	"Оптимизация" Операционной Системы / Отключить SWAP		free -h
2. Проверка соответствия программных требований к системе			
2.1 Установлена корректная версия и разрядность операционной системы (Ubuntu Server 22.04.x Astra Linux 1.8.x)			hostnamectl
2.2 Выставлено корректное имя хоста hostname: ecss1 или ecss2	Настройка /etc/hosts		cat /etc/hosts, cat /etc/hostname, CoCon команда: `/system/clusters/list`

Шаг	Пункт документации	Результат	Примечание
2.3 В системе установлены актуальные обновления	Обновление программного обеспечения операционной системы		
2.4 В системе установлено рекомендуемое ПО	Установка и настройка программного обеспечения		
2.5 В системе добавлен репозиторий ECSS-10 и имеется доступ к нему	Обновление программного обеспечения операционной системы		
2.6 В системе корректно настроены сетевые интерфейсы:	Настройка сети		
- *адреса сетевых интерфейсов статичны, не изменяются динамически по DHCP*	Настройка сети		sudo ifconfig
- *сервис keepalived (VRRP) настроен и работает*	Настройка VRRP для SIP-адаптера		При использовании резервирования sudo ifconfig
2.7 Ключ eToken/RuToken подключен к USB-порту сервера:	Проверка работы Token		
- *Ключ eToken/RuToken определился в системе (lsusb)*	Проверка работы Token		
- *Проверка ключа eToken/RuToken*	Проверка работы Token		
2.8 Проверка сетевой доступности между хостами ecss1 и ecss2:			При использовании резервирования
- *задержка ping между хостами не превышает 100 мс*			Необходимо минимизировать задержки трафика между хостами, насколько это возможно

Шаг	Пункт документации	Результат	Примечание

- отсутствуют потери пакетов при обмене данными между хостами			Запускается команда **ping** на несколько минут, после чего снимается статистика комбинацией клавиш:
			`CTRL+|`
			Пример:
			`ssw@ecss1:~$ ping -q -c 200 ecss2`
			`PING ecss2 (192.168.1.22) 56(84) bytes of data.`
			`177/177 packets, 0% loss, min/avg/ewma/max = 0.062/0.131/0.117/0.489 ms`

- хосты доступны между собой по DNS-имени (как прописано в /etc/hosts/ и в соответствии с лицензией)	Настройка /etc/hosts		Проверяется **ping** с ecss1 на ecss2 и наоборот

- между хостами настроена авторизация по ключам	Настройка /etc/hosts		Сгенерировать ssh-ключ и передать на другой сервер:
			`ssh-keygen`
			`ssh-copy-id ssw@ecss2`
			Аналогично с ecss2 на ecss1

2.9 На сервере работает синхронизация времени с NTP-сервером:	Установка ecss-node		Проверить статус: `ntpq -p`
- выставлен корректный часовой пояс	Установка часового пояса		Просмотреть: `date +%Z`
- на хостах ecss1 и ecss2 настроена синхронизация времени в режиме orphan либо синхронизация с единого NTP	Установка ecss-node		При использовании резервирования

Шаг	Пункт документации	Результат	Примечание
2.10. Установлен glusterfs-server:	Установка остальных пакетов ecss и их настройка/ glusterfs		Не ниже версии ПО 3.13.2
3. Проверка корректности установки ECSS-10			
3.1 Установка пакетов компонентов системы завершилась успешно без ошибок:			`dpkg -l | grep ecss`
- *ecss-postgres-bdr-ssw*	Инсталляция обязательных пакетов ECSS/ Установка Postgres		
- *ecss-node*	Инсталляция обязательных пакетов ECSS/ Установка ecss-node		
- *ecss-media-resources*	Инсталляция обязательных пакетов ECSS/ Установка остальных пакетов ecss и их настройка/ecss-media-server		
- *ecss-media-server*	Инсталляция обязательных пакетов ECSS/ Установка остальных пакетов ecss и их настройка/ecss-media-server		
- *ecss-restfs*	Инсталляция обязательных пакетов ECSS/ Установка остальных пакетов ecss и их настройка/ Установка пакета ecss-restfs и его настройка		

Шаг	Пункт документации	Результат	Примечание
- *ecss-user*	Инсталляция обязательных пакетов ECSS/ Установка ecss-node		
- *ecss-web-conf*	Инсталляция обязательных пакетов ECSS/ Установка остальных пакетов ecss и их настройка/ecss-web-conf		
- *ecss-dns-env*	Инсталляция обязательных пакетов ECSS/ Установка пакета ecss-dns-env		
4. Проверка работоспособности системы ECSS-10			
4.1 Проверка установленной лицензии ECSS-10:			CoCon команда: `/cluster/storage/ds1/licence/list-licence`
- *сервис pcscd находится в работе*	Установка ПО и подключение Token		`pkcs11-tool --module $(find /usr/lib/ecss/ecss-ds/lib/ -name librtpkcs11ecp.so | head -n1) -L`
- *в системе установлен корректный паспорт*			CoCon команда: `/cluster/storage/ds1/licence/show-passport`

Шаг	Пункт документации	Результат	Примечание
- в системе установлена актуальная лицензия			CoCon команда: `/cluster/storage/ds1/licence/current-limits`
4.2 Сервис dnsmasq запущен и работает:			

- если система с резервом, в конфигурации dnsmasq прописаны корректные адреса primary.broker.ecss и secondary.broker.ecss			Обратить внимание, что на обоих серверах содержимое одинаковое.
			`address=/primary.broker.ecss/<Адрес ecss1>`
			`address=/secondary.broker.ecss/<Адрес ecss2>`

- адреса доступны по DNS-имени	Инсталляция обязательных пакетов ECSS/ Установка пакета ecss-dns-env		`ping -c1 cocon.mysql.ecss`
			`ping -c1 dialer.mysql.ecss`
			`ping -c1 statistics.mysql.ecss`
			`ping -c1 tc.mysql.ecss`
			`ping -c1 tts.mysql.ecss`
			`ping -c1 system.restfs.ecss`

4.5 В системе с резервом прописано имя кластера (не undefined)			`cat /etc/ecss/ecss-mycelium/mycelium1.config | grep name`
4.6 Сервисы ecss запущенны и работают (находятся в состоянии running): • ecss-mycelium • ecss-core • ecss-ds • ecss-pa-sip • ecss-media-server • ecss-web-conf • ecss-restfs			`systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium`

Шаг	Пункт документации	Результат	Примечание
- Команда cocon "system-status" выдает пустой Alarms list			
4.7 Медиасервер MSR настроен и подключен к ECSS-10:	MSR Настройка медиасервера		
- конфигурация MSR настроена на подключение к ECSS-10	Настройка конфигурационного файла MSR		Из shell-серверов: `cat /etc/ecss/ecss-media-server/config.xml`

- MSR задекларирован в системных медиаресурсах	Команды управления медиаресурсами		CoCon:
			`/system/media/resource/list`
			`/system/media/registrar/info`
			Если система с резервированием, проверить, что подключен к обоим ядрам. Обратить внимание на правильные адреса ядер (не 0.0.0.0)

- На хосте ecss2 аналогичным образом сконфигурирован MSR			При использовании резервирования
4.8 Restfs настроен и доступен с сервера ECSS-10:			

- RestFS доступен для записи и скачивания медиафайлов			Из shell-серверов:
			`wget http://ecss1:9990/system/sounds/ai_you.wav`
			`wget http://ecss2:9990/system/sounds/ai_you.wav`

4.9 Сервис TTS (преобразование текста в голосовое сообщение) настроен и работает:			Если выбран tts при установке ecss-restfs

Шаг	Пункт документации	Результат	Примечание

- Сервис возвращает голосовой файл после преобразования из текста			Из shell-серверов:
			`wget http://ecss1:9990/generate?key=<Key>&text=<Text>&format=wav&lang=ru-RU&speaker=alyss&emotion=good&quality=hi`
			`wget http://ecss2:9990/generate?key=<Key>&text=<Text>&format=wav&lang=ru-RU&speaker=alyss&emotion=good&quality=hi`
			где:
			key=<Key> — ключ для доступа к серверу tts;
			text=<Text> — текстовое сообщение для преобразования в голосовое сообщение

4.10 Настроены пользователи, их права и роли	Управление пользователями		`cocon/list users`
4.11 Настроены правила ограничения обслуживания абонентов	Тип доступа, режим обслуживания, категория доступа и барринги		
5. Проверка корректности настройки домена в ECSS-10	Создание и настройка доменов		
5.1 В домене настроен SIP-транспорт (ip-set)	Работа SIP-адаптера с сетью		
- Адреса (node_ip) и порты (listen_port) выставлены в соответствии с проектом			`/cluster/adapter/sip1/sip/network/info`
- Адреса node_ip назначены на обеих нодах SIP-адаптера ecss1 и ecss2			При использовании резервирования

Шаг	Пункт документации	Результат	Примечание
- Адреса node_ip совпадают с адресами резервируемых интерфейсов, настроенных в keepalived			При использовании резервирования
- На домене выставлен ipset			`/domain/<DOMAIN>/sip/network/info`
5.2 В домен добавлены необходимые услуги:	Инсталляция и управление услугами		
- произведена установка в систему услуг	Инсталляция и управление услугами		`cluster/storage/ds1/ss/install ds1@ecss1 *`

- домен добавлен в access-list к услугам	Инсталляция и управление услугами		Просмотреть установленные в доменах услуги:
			`/cluster/storage/ds1/ss/access-list show`
			Для добавления услуг на домене:
			`/cluster/storage/ds1/ss/access-list add <DOMAIN> <SS>`

- настроена система сбора CDR (при необходимости)	Управление системой сбора CDR		
5.3 Произведена настройка контекста маршрутизации:	Виртуальная АТС. Маршрутизация телефонных вызовов		`/domain/<DOMAIN>/routing/list` `/domain/<DOMAIN>/routing/show <CTX>`

Шаг	Пункт документации	Результат	Примечание

- настроены нужные транки	Команды управления SIP-транками		При необходимости.
			`/domain/<DOMAIN>/trunk/sip/info`
			`/domain/<DOMAIN>/trunk/info`

- настроены нужные бриджи	Команды управления bridge-интерфейсами		При необходимости. `/bridge/info`
- маршруты отрабатываются как в режиме набора enblock, так и overlap	Команды управления маршрутизацией		`/domain/<DOMAIN>/routing/trace`
5.4 Произведены корректные настройки SIP-абонента(-ов):	Виртуальная АТС. Подключение и настройка SIP-абонентов		
- абоненты созданы	Создание SIP-абонента		`domain/<DOMAIN>/sip/user/list`

- установлены надежные логин/пароль SIP-авторизации			`domain/<DOMAIN>/sip/user/info * *`
			или отфильтровать (из shell):
			`ssh admin@ecss1 -p8023 '/domain/<DOMAIN>/sip/user/info * *' | grep '(ds)'`

- установлен trusted ip	Аутентификация абонентов		В случае необходимости SIP-регистрации без авторизации `domain/<DOMAIN>/sip/user/info`

Шаг	Пункт документации	Результат	Примечание

- у абонента установлен корректный контекст маршрутизации	Добавление абонента		`domain/<DOMAIN>/sip/user/info`
			или
			`cluster/storage/ds1/iface/ list <DOMAIN> * <GROUP> routing_context ~ ctx_from_local`
			(подставить нужный)

- выставлен корректный тип терминала (basic/smart)	Настройка параметров абонента		`/cluster/storage/ds1/iface/ list <DOMAIN> * <GROUP> terminal_type ~ smart` (или basic)
- у абонента активированы необходимые услуги	Инсталляция и управление услугами		`/domain/<DOMAIN>/ss/info`
5.5 Установка контекстов маршрутизации для системных интерфейсов:	Команды управления системными интерфейсами		`/domain/<DOMAIN>/iface/user-set .system .system system:ivr routing.context`
- настроен system:ivr	Команды управления системными интерфейсами		При необходимости использования сервиса IVR. В большинстве случаев прописывается только контекст маршрутизации `/domain/<DOMAIN>/iface/ info .system .system system:ivr`

Шаг	Пункт документации	Результат	Примечание
- *настроен system:teleconference*	Команды управления системными интерфейсами		При необходимости использования сервиса Teleconference. В большинстве случаев прописывается только контекст маршрутизации: `/domain/<DOMAIN>/iface/info .system .system system:teleconference`

16 Регламент обновления/деинсталляции ECSS

Обновление с версии 3.14.14 на 3.14.15

Обновление с версии 3.14.15 на 3.14.16 Ubuntu

Обновление с версии 3.14.15 на 3.14.16 Astra

Обновление пакета ecss-postgres-bdr-ssw

16.1 Обновление пакета ecss-postgres-bdr-ssw

В текущей версии, при необходимости обновления версии пакета **ecss-postgres-bdr-ssw** необходимо выполнить следующие действия:

- Удалить установленный пакет **ecss-postgres-bdr-ssw** командой:

на обоих хостах

```
sudo apt remove ecss-postgres-bdr-ssw
```

- Остановить docker контейнер

на обоих хостах

```
sudo docker compose -f /srv/ecss/ecss-postgres-bdr-ssw/docker-compose.container.yml down
```

- Установить новую версию пакета **ecss-postgres-bdr-ssw** командой:

на обоих хостах

```
sudo apt install ecss-postgres-bdr-ssw
```

Пример:

Было:

```
dpkg -l|grep ecss-postgres
ii  ecss-postgres-bdr-ssw                        17.1.0.55+ssw
amd64        Eltex/Postgres BDR

sudo docker ps
CONTAINER ID   IMAGE                                                COMMAND
CREATED       STATUS       PORTS                                          NAMES
00b8a991de94   git.eltex.loc:4567/ecss/ecss-postgresbdr:3.17   "/docker-
```

```
entrypoint.…"   2 weeks ago   Up 2 hours   0.0.0.0:5439->5432/tcp, [::]:5439-
>5432/tcp   ecss-postgres-bdr-ssw-postgres-container-1
```

Удаляем текущую версию пакета ecss-postgres-bdr-ssw

```
sudo apt remove ecss-postgres-bdr-ssw
… … …
Удаляется ecss-postgres-bdr-ssw (17.1.0.55+ssw) …
```

Удаляем docker контейнер

```
 sudo docker compose -f /srv/ecss/ecss-postgres-bdr-ssw/docker-
compose.container.yml down
[+] down 1/2
 ✓ Container ecss-postgres-bdr-ssw-postgres-container-1 Removed
10.8s
 ⠟ Network ecss-postgres-bdr-ssw_default                Removing
```

Устанавливаем новую версию пакета ecss-postgres-bdr-ssw

```
sudo apt install ecss-postgres-bdr-ssw
… … …
Распаковывается ecss-postgres-bdr-ssw (17.1.0.59+ssw) …
Настраивается пакет ecss-postgres-bdr-ssw (17.1.0.59+ssw) …

======================
 Loading Docker image
======================
Loaded image: git.eltex.loc:4567/ecss/ecss-postgresbdr:3.17

======================
 Starting Postgres DB
======================
[+] up 2/2
 ✓ Network ecss-postgres-bdr-ssw_default                Created
0.1s
 ✓ Container ecss-postgres-bdr-ssw-postgres-container-1 Created
0.2s

==========================
 Waiting for PostgreSQL DB
==========================
Waiting for PostgreSQL (attempt 1)...
Waiting for PostgreSQL (attempt 2)...
Waiting for PostgreSQL (attempt 3)...
Waiting for PostgreSQL (attempt 4)...
Waiting for PostgreSQL (attempt 5)...
```

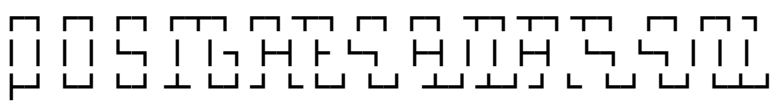
```
================================
 Creating Postgres Exporter Role
================================
Node IP: localhost
DO
ALTER ROLE
GRANT

=================
 Create databases
=================
Database ecss_storekeeper_db already exists, skipping...
Database web_conf already exists, skipping...
Extension btree_gist already exists, skipping...
Extension bdr already exists, skipping...

==================
 Joining BDR group
==================
This node is already part of the BDR group, skipping join...

============================
 Checking replication status
============================
Репликация BDR настроена корректно: данные совпадают.

=========================================================
=========================================================
 ecss-postgres-bdr-ssw successfully installed/updated!
=========================================================
=========================================================
```



Результат:

```
dpkg -l|grep ecss-postgres
ii  ecss-postgres-bdr-ssw                        17.1.0.59+ssw
amd64        Eltex/Postgres BDR


sudo docker ps
CONTAINER ID   IMAGE                                                COMMAND
CREATED         STATUS         PORTS                                                 NAMES
2a8ca5e72920   git.eltex.loc:4567/ecss/ecss-postgresbdr:3.17   "/docker-entrypoint.…"   8
 minutes ago   Up 8 minutes   0.0.0.0:5439->5432/tcp, [::]:5439->5432/tcp   ecss-postgres-
bdr-ssw-postgres-container-1
```

16.2 Обновление с версии 3.14.14 на 3.14.15

16.2.1

- Обновление с версии 3.14.14 на 3.14.15
  - Подготовка к обновлению системы
  - Установка ECSS-10 версии 3.14.15
    - Установка ecss-mysql
    - Установка ecss-node
    - Установка остальных пакетов
- Обновление второго сервера, если система в кластере
  - Установка ecss-mysql
  - Установка ecss-node
  - Установка остальных пакетов
  - Сборка кластера
  - Проверка работы системы в кластере
- Деинсталляция компонентов ECSS
  - Удаление ecss-mysql
  - Удаление пакета ecss-node
  - Удаление пакета ecss-media-server
  - Удаление ненужных пакетов
  - Удаление пакетов по маске
  - Команды полной деинсталляции ECSS
  - Debconf

16.2.2 Обновление с версии 3.14.14 на 3.14.15

Рекомендуется производить обновление с новой установки операционный системы (Ubuntu Server 18.04.x LTS 64bit) и новой разметки дискового пространства.

В данном разделе описан процесс обновления операционной системы и программных компонентов ECSS-10, когда по каким-либо причинам новая установка невозможна.

В версии 3.14.15 изменился алгоритм проверки валидности паспорта лицензий. В предыдущих версиях допускался паспорт без окончания срока действия, сейчас такое недопустимо. Во избежание проблем после обновления, рекомендуется на первом этапе проверить параметры паспорта следующей командой:

```
/cluster/storage/ds1/licence/show-passport
SSW ID              ECSS XXXXXX
Description         Элтекс,
Create date         27.12.2024 07:07:09
. . .
Expired date        31.12.2025 23:59:59
. . .
```

В случае отсутствия строчки Expired date (паспорт без окончания срока действия), до начала обновления необходимо обратится в коммерческий отдел Элтекс для обновления паспорта и лицензии.

Подготовка к обновлению системы

Выполнить полный «бэкап» (см. «Регламентное обслуживание и резервное копирование»).

Если нужны последние данные биллинга, то необходимо финализировать текущую cdr и скачать в надежное место, например:

```
/domain/bsk.test/cdr/make_finalize_cdr bsk3
Created file(s): cdrbsk3_20210330_11_05_38_t.csv
```

Посмотреть, сколько зарегистрированных абонентов, пример:

```
/domain/bsk.test/sip/user/registered *
....
┌──────────────┐
│ elements: 320 │
└──────────────┘
```

При обновлении кластера следует обновлять сервера по очереди. Таким образом, при обновлении один из серверов будет всегда запущен и работа сервисов не будет прервана. Для разделения кластера следует использовать инструмент ecss-control на обоих серверах (Описание команд утилиты приведено в «Приложении Н»).

Переведите трафик на второй сервер, выключив keepalive на первом хосте:

на ecss1

```
sudo systemctl stop keepalived.service
```

Командой ifconfig проверьте, что все IP-адреса находятся на ecss2.

Выполните разделение SSW на два независимых сервера следующей командой:

Команда **ecss-control** работает со стандартными именами хостов : **ecss1**/**ecss2**.

на ecss1

```
sudo ecss-control detach
```

```
sudo ecss-control detach
detach host: ecss2, ip: 10.10.0.2
complete
```

```
sudo ecss-control status
locked ip: 10.10.0.2
```

После разделения кластера необходимо выключить все ноды на сервере, который будет обновляться первым:

на ecss1

```
sudo systemctl stop ecss-mediator.service
sudo systemctl stop ecss-pa-sip.service
sudo systemctl stop ecss-core.service
sudo systemctl stop ecss-ds.service
sudo systemctl stop ecss-mycelium.service
```

Выполнить обновление:

на ecss1

```
sudo apt update
```

Установка ECSS-10 версии 3.14.15

Подробнее в разделе «Инсталляция пакетов ECSS».

Установка ecss-mysql

Обновите ecss-mysql на первом хосте:

на ecss1

```
sudo apt install ecss-mysql
```

Если возникла проблема с какой-либо базой данных, следует произвести перезагрузку сервиса aparmor, затем удалить базу данных из mysql, а после снова установить ecss-mysql:

```
sudo systemctl restart apparmor.service
mysql -uroot -p<root_mysql_password> -e 'DROP DATABASE <проблемная база данных>'
sudo apt install ecss-mysql
```

- Произведите мониторинг состояния реплики с помощью SHOW SLAVE STATUS; и дождитесь, пока показатель Seconds_Behind_Master станет равным "0";

Установка ecss-node

Установить пакет ecss-node:

на ecss1

```
sudo apt install ecss-node
```

В процессе установки будут задаваться вопросы о сохранении текущих версий конфигурационных файлов. Рекомендуется ставить новые, т. е. отвечать «Y».

на ecss1

во время установки пакета ecss-node , система сбросит конфигурацию кластера на дефолтное значение. Поэтому потребуется отредактировать файл /etc/ecss/ecss-mycelium/mycelium1.config.
выполните команду :

на ecss1

```
sudo nano /etc/ecss/ecss-mycelium/mycelium1.config
```

пропишите имя кластера, в примере "**test_cluster**"

```
%%% -*- mode:erlang -*-
%%% Warning - this config file *must* end with <dot><whitespace>
[
    {mycelium_broker, [
        {cluster_name, test_cluster},
        %{cluster_nodes, []}, % additional nodes
        %{tcp_port, 5672},
        %{auto_discoverer, true},
        {compress_module, elx_data_nif}, %% elx_data_nif | elx_data
        {session_gc_timeout, 3000},
        {queue_gc_timeout, infinity}
        %{info_realm, {local, "/ssw", "ssw", "ssw", 1, [{localhost, 0, {host,
"localhost"}, 5672, 10}]}},
        %{bind_ip_addresses, [{127,0,0,1}]} % do not forget to include lo
interface
        ]},
```

Установка остальных пакетов

Установить пакеты, содержащие основные подсистемы:

на ecss1

```
sudo apt install ecss-restfs
```

на ecss1

```
sudo apt install ecss-media-server
```

на ecss1

```
sudo apt install ecss-media-resources
```

на ecss1

```
sudo apt install ecss-web-conf
```

- При установке ecss-media-server рекомендуется ответить на вопрос о конфигурировании утвердительно.

- В процессе установки будут задаваться вопросы о сохранении текущих версий конфигурационных файлов. Рекомендуется ставить новые, т. е. отвечать «Y».

Запустите сервисы следующими командами:

на ecss1

```
sudo systemctl start ecss-mycelium.service
```

на ecss1

```
sudo systemctl start ecss-ds.service
```

на ecss1

```
sudo systemctl start ecss-core.service
```

на ecss1

```
sudo systemctl start ecss-pa-sip.service
```

на ecss1

```
sudo systemctl start ecss-mediator.service
```

проверить статус сервисов командой:

на ecss1

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
```

Проверить состояние сервисов в CoCon:

выполняется в CoCon

```
/system-status
Checking...
┌─┬─────────────────┬──────────────────────────┬─────────────────┬─────────────┬───────┐
│ │      Node       │         Release          │  Erlang nodes   │Mnesia nodes│Uptime│
├─┼─────────────────┼──────────────────────────┼─────────────────┼─────────────┼───────┤
│ │core1@ecss1      │ecss-core-3.14.15.1163    │core1@ecss1      │not running │3m 45s│
│ │ds1@ecss1        │ecss-ds-3.14.15.1163      │ds1@ecss1        │ds1@ecss1    │3m 51s│
│ │md1@ecss1        │ecss-mediator-3.14.15.1163│md1@ecss1        │md1@ecss1    │3m 32s│
│ │mycelium1@ecss1  │ecss-mycelium-3.14.15.1163│mycelium1@ecss1  │not running │3m 58s│
│ │sip1@ecss1       │ecss-pa-sip-3.14.15.1163  │sip1@ecss1       │sip1@ecss1   │3m 39s│
└─┴─────────────────┴──────────────────────────┴─────────────────┴─────────────┴───────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────┬────────────┬────────────┐
│     Node     │    MSR   │    MSR   │ Cc-status  │ Cc-uptime  │
│              │          │  version │            │            │
├──────────────┼──────────┼──────────┼────────────┼────────────┤
│ core1@ecss1  │ msr.ecss1 │ 3.14.15.221 │ connected │ 00:02:28  │
│              │ msr.ecss2 │ 3.14.14.24  │ connected │ 00:02:28  │
└──────────────┴──────────┴──────────┴────────────┴────────────┘
```

В версии 3.14.15 появилась возможность использовать любые имена хостов (стандартные имена отсутствуют), поэтому нужно указать какие имена мы будем использовать, для этого выполните следующую команду (в примере имена хостов **ecss1**/**ecss2**):

выполняется в CoCon

```
/system/clusters/set [ecss1, ecss2]
```

```
/system/clusters/set [ecss1, ecss2]
Success: System initialized on hosts: ["ecss2","ecss1"]


/system/clusters/list
┌─────────┬──────────────┬──────────────┐
│  Role   │    Cluster   │     Nodes    │
├─────────┼──────────────┼──────────────┤
│storage  │ds1           │ds1@ecss2     │
```

```
|         |               |ds1@ecss1     |
|core     |core1          |core1@ecss2   |
|         |               |core1@ecss1   |
|adapter  |sip1           |sip1@ecss2    |
|         |               |sip1@ecss1    |
|adapter  |megaco1        |megaco1@ecss2 |
|         |               |megaco1@ecss1 |
|adapter  |sigtran1       |sigtran1@ecss2|
|         |               |sigtran1@ecss1|
|adapter  |sorm1          |sorm1@ecss2   |
|         |               |sorm1@ecss1   |
|mediator|md1             |md1@ecss2     |
|         |               |md1@ecss1     |
|virtual  |.system_bridge|               |
|virtual  |.virtual       |              |
└─────────┴───────────────┴──────────────┘
```

В случае использования для сигнальной сети нескольких vlan-ов , необходимо скорректировать конфигурационный файл **/etc/ecss/ecss-media-server/config.xml** описав параметр "via":
via — задается список интерфейсов, которые могут быть отправлены в сообщении "SIP REGISTER" поле "Via":

- dynamic-ifaces — значение = "yes" или "no";
  если dynamic-ifaces="no":
  - при мастер-регистрации всё как раньше
  - в ином случае регистрация проходит ТОЛЬКО если интерфейс перечислен в списке via в конфиге
  если dynamic-ifaces="yes" или атрибут не указан, то остаётся поведение, реализованное ранее.
- iface name — имя интерфейса;
- ip — ip адрес интерфейса.
  - если указанный в конфиге в разделе via ip-адрес не существует, он не обрабатывается (пропускается)
  - если указан верный ip, то он прикрепляется для регистрации в заголовок Via
  - при отсутствии интерфейса в конфиге в via при мастер-регистрации указывается bind-address из конфига, при при не-мастер регистрации поведение прежнее.

пример для одиночного хоста

```
<?xml version="1.0" encoding="utf-8"?>
<config date="14:39:00 10.03.2023">
  <general log-level="3" log-rotate="yes" max-calls="8192" max-in-group="512" load-
sensor="media" load-delta="10" calls-delta="100" spool-dir-size="100M" log-name="msr.log
" log-path="/var/log/ecss/media-server" use-srtp="disabled" enable-ice-transport="no"
 ice-update="no" aggressive-ice="yes" stun-server="" suspicious-mode="no"/>
  <transport bind-addr="192.168.1.40" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca" user="webrtc@live.com" password="mu
azkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60" rtcp-timeout="0
" rtp-timeout="350" udp-src-check="no" cn-multiplier="3" port-start="12000" port-range="
2048" tias-in-sdp="no" thread-cnt="2" vid-enc-threads="2" vid-dec-threads="2" video-
conf-layout="evenly" keyframe-interval="1000" vid-decode-delay="100" silence-threshold="
-30" dtmf-flash-disable="no" video-dscp="0" other-dscp="0" dummy-video-src="/usr/share/
ecss-media-server/video/dummy_video.yuv" video-enc-width="640" video-enc-height="360"
 finalsilence="1000" rtcp-stat-dump="yes"/>
  <codec pcma="1" pcmu="2" ilbc="0" gsm="0" g722="3" g729="0" speex="0" l16="0" g7221="0
" opus="0" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
  <accounts>
     <dynamic msr_name="msr.name" realm="sip:192.168.1.40:5000" dtmf_mode="rfc+inband+in
fo" auth_name="user" auth_password="password">
       <via dynamic-ifaces="no">
          <iface name="bond1.75" ip="192.168.1.27"/>
          <iface name="bond1.20" ip="192.168.1.28"/>
          <iface name="bond1.35" ip="192.168.1.29"/>
          <iface name="bond1.36" ip="192.168.1.30"/>
       </via>
     </dynamic>
  </accounts>
  <pbyte>
    <mcc bind-addr="192.168.1.40" port="5700"/>
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
```

```
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

пример для кластерной конфигурации

```
<?xml version="1.0" encoding="utf-8"?>
<config date="14:39:00 10.03.2023">
  <general log-level="3" log-rotate="yes" max-calls="8192" max-in-group="512" load-
sensor="media" load-delta="10" calls-delta="100" spool-dir-size="100M" log-name="msr.log
" log-path="/var/log/ecss/media-server" use-srtp="disabled" enable-ice-transport="no"
 ice-update="no" aggressive-ice="yes" stun-server="" suspicious-mode="no"/>
  <transport bind-addr="192.168.1.40" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca" user="webrtc@live.com" password="mu
azkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60" rtcp-timeout="0
" rtp-timeout="350" udp-src-check="no" cn-multiplier="3" port-start="12000" port-range="
2048" tias-in-sdp="no" thread-cnt="2" vid-enc-threads="2" vid-dec-threads="2" video-
conf-layout="evenly" keyframe-interval="1000" vid-decode-delay="100" silence-threshold="
-30" dtmf-flash-disable="no" video-dscp="0" other-dscp="0" dummy-video-src="/usr/share/
ecss-media-server/video/dummy_video.yuv" video-enc-width="640" video-enc-height="360"
 finalsilence="1000" rtcp-stat-dump="yes"/>
  <codec pcma="1" pcmu="2" ilbc="0" gsm="0" g722="3" g729="0" speex="0" l16="0" g7221="0
" opus="0" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
  <accounts>
     <dynamic msr_name="msr.name" realm="sip:192.168.1.40:5000" dtmf_mode="rfc+inband+in
fo" auth_name="user" auth_password="password">
       <via dynamic-ifaces="no">
          <iface name="bond1.75" ip="192.168.1.27"/>
          <iface name="bond1.20" ip="192.168.1.28"/>
          <iface name="bond1.35" ip="192.168.1.29"/>
          <iface name="bond1.36" ip="192.168.1.30"/>
       </via>
     </dynamic>
     <dynamic msr_name="msr.name" realm="sip:192.168.1.50:5000" dtmf_mode="rfc+inband+in
fo" auth_name="user" auth_password="password">
       <via dynamic-ifaces="no">
          <iface name="bond1.75" ip="192.168.1.27"/>
          <iface name="bond1.20" ip="192.168.1.28"/>
          <iface name="bond1.35" ip="192.168.1.29"/>
          <iface name="bond1.36" ip="192.168.1.30"/>
       </via>
     </dynamic>
  </accounts>
  <pbyte>
    <mcc bind-addr="192.168.1.40" port="5700"/>
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

Хост ecss1 обновлен , активизируем трафик командой:

на ecss1

```
sudo systemctl start keepalived.service
```

16.2.3 Обновление второго сервера, если система в кластере

Чтобы перевести трафик на обновленный сервер, нужно на втором сервере остановить службу keepalived:

на ecss2

```
sudo systemctl stop keepalived.service
```

Если нужны последние данные биллинга, то необходимо финализировать текущую cdr и скачать в надежное место, например:

```
admin@mycelium1@ecss1:/$ domain/bsk.test/cdr/make_finalize_cdr bsk3
Created file(s): cdrbsk3_20220128_10_06_40_t.csv
```

Выключить все ноды на сервере, который будет обновляться вторым:

на ecss2

```
sudo systemctl stop ecss-mediator.service
sudo systemctl stop ecss-pa-sip.service
sudo systemctl stop ecss-core.service
sudo systemctl stop ecss-ds.service
sudo systemctl stop ecss-mycelium.service
```

Выполнить обновление:

на ecss2

```
sudo apt update
```

Установка ecss-mysql

Обновите ecss-mysql на втором хосте:

на ecss2

```
sudo apt install ecss-mysql
```

Если возникла проблема с какой-либо базой данных, следует произвести перезагрузку сервиса aparmor, затем удалить базу данных из mysql, а после снова установить ecss-mysql:

```
sudo systemctl restart apparmor.service
mysql -uroot -p<root_mysql_password> -e 'DROP DATABASE <проблемная база данных>'
sudo apt install ecss-mysql
```

- Произведите мониторинг состояния реплики с помощью SHOW SLAVE STATUS; и дождитесь, пока показатель Seconds_Behind_Master станет равным "0";

Установка ecss-node

Установить пакет ecss-node:

на ecss2

```
sudo apt install ecss-node
```

В процессе установки будут задаваться вопросы о сохранении текущих версий конфигурационных файлов. Рекомендуется ставить новые, т. е. отвечать «Y».

на ecss2

Во время установки пакета ecss-node система сбросит конфигурацию кластера на дефолтное значение. Поэтому потребуется отредактировать файл /etc/ecss/ecss-mycelium/mycelium1.config.
Выполните команду:

```
sudo nano /etc/ecss/ecss-mycelium/mycelium1.config
```

Пропишите имя кластера, в примере "**test_cluster**":

```
%%% -*- mode:erlang -*-
%%% Warning - this config file *must* end with <dot><whitespace>
[
    {mycelium_broker, [
        {cluster_name, test_cluster},
        %{cluster_nodes, []}, % additional nodes
        %{tcp_port, 5672},
        %{auto_discoverer, true},
        {compress_module, elx_data_nif}, %% elx_data_nif | elx_data
        {session_gc_timeout, 3000},
        {queue_gc_timeout, infinity}
        %{info_realm, {local, "/ssw", "ssw", "ssw", 1, [{localhost, 0, {host,
"localhost"}, 5672, 10}]}},
        %{bind_ip_addresses, [{127,0,0,1}]} % do not forget to include lo
interface
        ]},
```

Установка остальных пакетов

Установите пакеты, содержащие основные подсистемы:

на ecss2

```
sudo apt install ecss-restfs
```

на ecss2

```
sudo apt install ecss-media-server
```

на ecss2

```
sudo apt install ecss-media-resources
```

на ecss2

```
sudo apt install ecss-web-conf
```

При установке ecss-media-server рекомендуется ответить на вопрос о конфигурировании утвердительно.

В процессе установки будут задаваться вопросы о сохранении текущих версий конфигурационных файлов. Рекомендуется ставить новые, т. е. отвечать «Y».

Проверьте статус сервисов командой:

на ecss2

```
systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
```

Сервисы должны быть в статусе "inactive":

/systemctl is-active ecss-core ecss-pa-sip ecss-ds ecss-mediator ecss-mycelium
inactive
inactive
inactive
inactive
inactive

Сборка кластера

После действий в предыдущем разделе можно открывать серверы друг другу, чтобы свести в кластер. На первом сервере выполните команду **sudo ecss-control attach**. Описание команд утилиты приведено в «Приложении Н».

Пример:

на ecss1

```
sudo ecss-control attach
unlock ip: 10.10.0.2, rules: 7
complete
sudo ecss-control status
all nodes attached
```

Запустите сервисы следующими командами:

на ecss2

```
sudo systemctl start ecss-mycelium.service
```

на ecss2

```
sudo systemctl start ecss-ds.service
```

на ecss2

```
sudo systemctl start ecss-core.service
```

на ecss2

```
sudo systemctl start ecss-pa-sip.service
```

на ecss2

```
sudo systemctl start ecss-mediator.service
```

Проверка работы системы в кластере

Проверьте выполнение обновления. Команда CLI **system-status** должна показать, что все ноды в кластере, версии подсистем одинаковые.

выполняется в CoCon

```
/system-status
Checking...
┌─┬──────────────┬──────────────────────────┬──────────────────────────────────────────┬──────────────┬──────┐
│ │     Node     │         Release          │              Erlang nodes                │ Mnesia nodes │Uptime│
├─┼──────────────┼──────────────────────────┼──────────────────────────────────────────┼──────────────┼──────┤
│ │core1@ecss1   │ecss-core-3.14.15.1163    │core1@ecss1,core1@ecss2                   │not running   │1h 5m │
│ │core1@ecss2   │ecss-core-3.14.15.1163    │core1@ecss1,core1@ecss2                   │not running   │3m 8s │
│ │ds1@ecss1     │ecss-ds-3.14.15.1163      │ds1@ecss1,ds1@ecss2                       │ds1@ecss1,ds1@ecss2  │1h 5m │
│ │ds1@ecss2     │ecss-ds-3.14.15.1163      │ds1@ecss1,ds1@ecss2                       │ds1@ecss1,ds1@ecss2  │59s   │
│ │md1@ecss1     │ecss-mediator-3.14.15.1163│md1@ecss1,md1@ecss2                       │md1@ecss1,md1@ecss2  │1h 5m │
│ │md1@ecss2     │ecss-mediator-3.14.15.1163│md1@ecss1,md1@ecss2                       │md1@ecss1,md1@ecss2  │2m 53s│
│ │mycelium1@ecss1│ecss-mycelium-3.14.15.1163│mycelium1@ecss1,mycelium1@ecss2          │not running   │1h 5m │
│ │mycelium1@ecss2│ecss-mycelium-3.14.15.1163│mycelium1@ecss1,mycelium1@ecss2          │not running   │3m 17s│
│ │sip1@ecss1    │ecss-pa-sip-3.14.15.1163  │sip1@ecss1,sip1@ecss2                     │sip1@ecss1,sip1@ecss2│1h 5m │
│ │sip1@ecss2    │ecss-pa-sip-3.14.15.1163  │sip1@ecss1,sip1@ecss2                     │sip1@ecss1,sip1@ecss2│2m 57s│
└─┴──────────────┴──────────────────────────┴──────────────────────────────────────────┴──────────────┴──────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────────┬────────────┬──────────────┐
│     Node     │    MSR   │     MSR      │ Cc-status  │ Cc-uptime    │
│              │          │   version    │            │              │
├──────────────┼──────────┼──────────────┼────────────┼──────────────┤
│ core1@ecss1  │ msr.ecss1 │ 3.14.15.221 │ connected  │ 01:04:13     │
│              │ msr.ecss2 │ 3.14.15.221 │ connected  │ 00:15:04     │
│ core1@ecss2  │ msr.ecss1 │ 3.14.15.221 │ connected  │ 00:01:59     │
│              │ msr.ecss2 │ 3.14.15.221 │ connected  │ 00:01:57     │
└──────────────┴──────────┴──────────────┴────────────┴──────────────┘
```

Хост ecss2 обновлен, запустите трафик командой:

на ecss2

```
sudo systemctl start keepalived.service
```

Убедитесь, что абоненты зарегистрированы. Проверьте количество зарегистрированных абонентов, оно должно быть приблизительно тем же, что и в начале обновления:

выполняется в CoCon

```
/domain/bsk.test/sip/user/registered *
320 make users list ...
[************************************************************************] 223mks
320 users information read ...
[************************************************************************] 22ms
Executed on the sip1@ecss1
┌──────────────────────┬────────┬─────────┬──────────────────────────────────────────────────────────────────────────────
──────────────────────┐
─────┐
│     Interface    │ Number │ Dynamic │                                  Contact
│                         Registration info                                  │
├──────────────────────┼────────┼─────────┼──────────────────────────────────────────────────────────────────────────────
──────────────────────┤
─────┤
...
┌─────────────────┐
│ elements: 320 │
└─────────────────┘
```

Вызовы проходят во всех направлениях, CFC проигрывает сообщения, Web-conf и прочие web-сервисы запускаются, в системный журнал не пишется ошибок. Команда **shell**:

```
journalctl -f
```

16.2.4 Деинсталляция компонентов ECSS

Удаление ecss-mysql

Для удаления пакета ecss-mysql с сохранением используйте команду:

```
sudo apt remove ecss-mysql
```

При этом базы и конфигурационные файлы не удаляются.

Для полного удаления пакета вместе с базами и конфигурационными файлами используйте ключ **purge**:

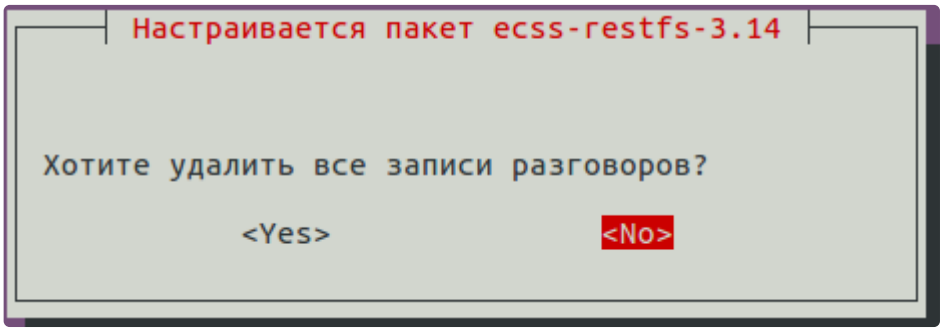
```
sudo apt purge ecss-mysql
```

При этом по каждой базе нужно подтверждение удаления. После этого каталог **/var/lib/ecss-mysql/** будет очищен.

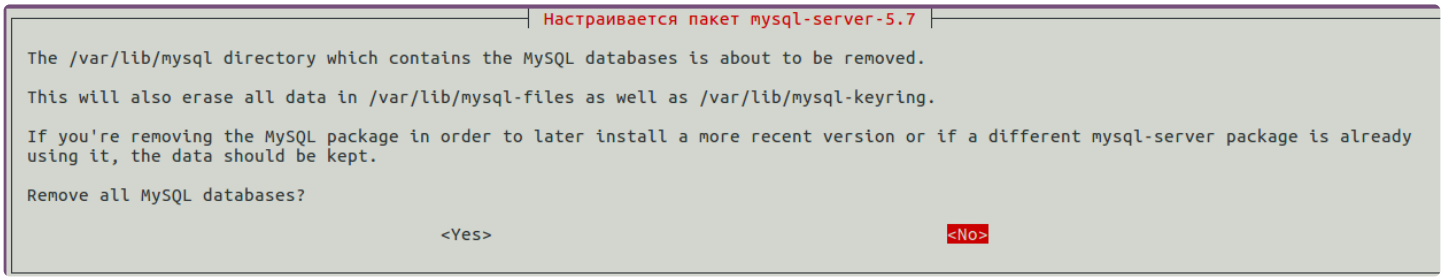
Если нужно полностью удалить все пакеты mysql* :

Удаление пакетов mysql-* удалит также пакеты ecss-web-conf и ecss-restfs.

При удалении ecss-restfs задается вопрос об удалении записей разговоров:



А также об удалении всех БД MySQL:




```
sudo apt purge mysql-*
```

Удаление пакета ecss-node

Для удаления пакета используйте команду:

```
sudo apt purge ecss-node
```

Удаление пакета ecss-media-server

Для удаления пакета с сохранением конфигурационных файлов используйте команду:

```
sudo apt remove ecss-media-server
```

При этом в каталоге **/etc/ecss/ecss-media-server/** остаются файлы конфигурации.

Для полного удаления используйте команду:

```
sudo apt purge ecss-media-server
```

Удаление ненужных пакетов

Для удаления остальных пакетов, которые больше не требуются, используйте ключ autoremove:

```
sudo apt autoremove
```

Удаление пакетов по маске

Для удаления сразу нескольких пакетов можно пользоваться масками (см. man apt).

Пример команды для удаления всех пакетов ecss:

```
sudo apt purge ecss-*
```

Команды полной деинсталляции ECSS

Для полного удаления всех компонентов ECSS выполните 3 команды:

```
sudo apt purge ecss-*
sudo apt purge mysql-*
sudo apt autoremove
```

После следует перезагрузить сервер для удаления некоторых процессов из оперативной памяти.

⚠ В каталоге **/var/log/ecss/security/** в служебных целях остаются логи. Их нельзя удалить штатными средствами. Если необходима очистка этого каталога, следует обратиться в техническую поддержку.

Debconf

В некоторых случаях вместо переустановки пакета требуется только изменение конфигурации. Здесь следует применять команду **dpkg-reconfigure <имя_пакета>**. Также при необходимости можно использовать команды пакета **debconf-utils**.

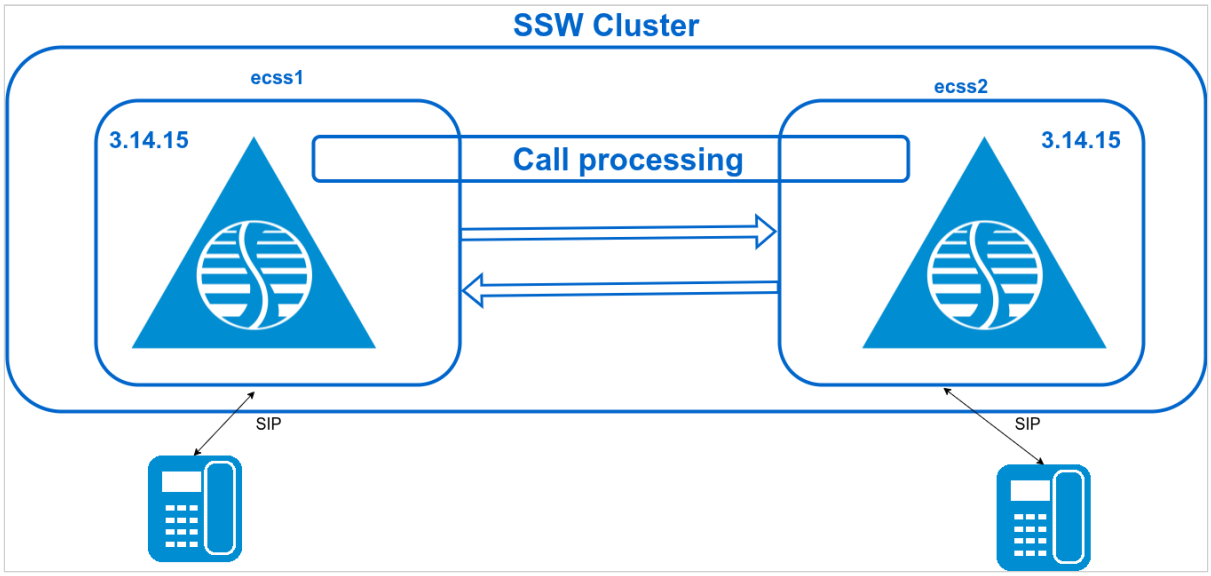
Описание приведено на странице «Приложение В».

16.3 Обновление с версии 3.14.15 на 3.14.16 Astra

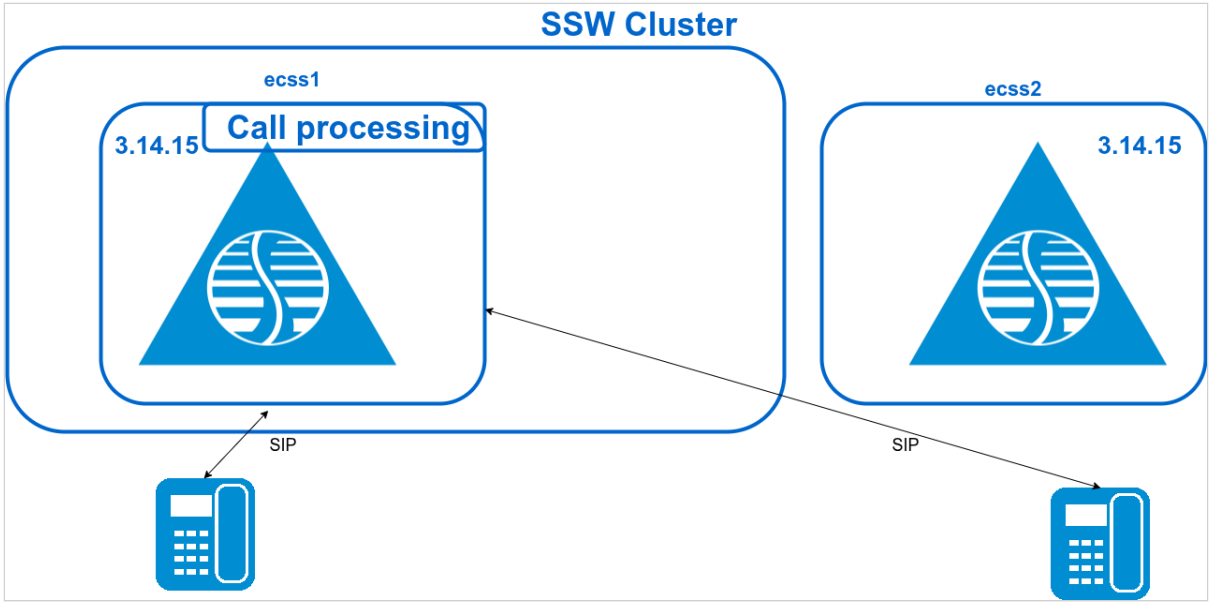
- Обновление с версии 3.14.15 на 3.14.16
  - 1 Подготовительные действия на ecss2
  - 2 Подготовка к обновлению сервисов SSW на ecss2
  - 3 Обновление сервисов SSW на ecss2
  - 4 Подготовительные действия на ecss1
  - 5 Подготовка к обновлению сервисов SSW на ecss1
  - 6 Обновление сервисов SSW на ecss1
  - 7 Проверка обновления

16.3.1 Обновление с версии 3.14.15 на 3.14.16

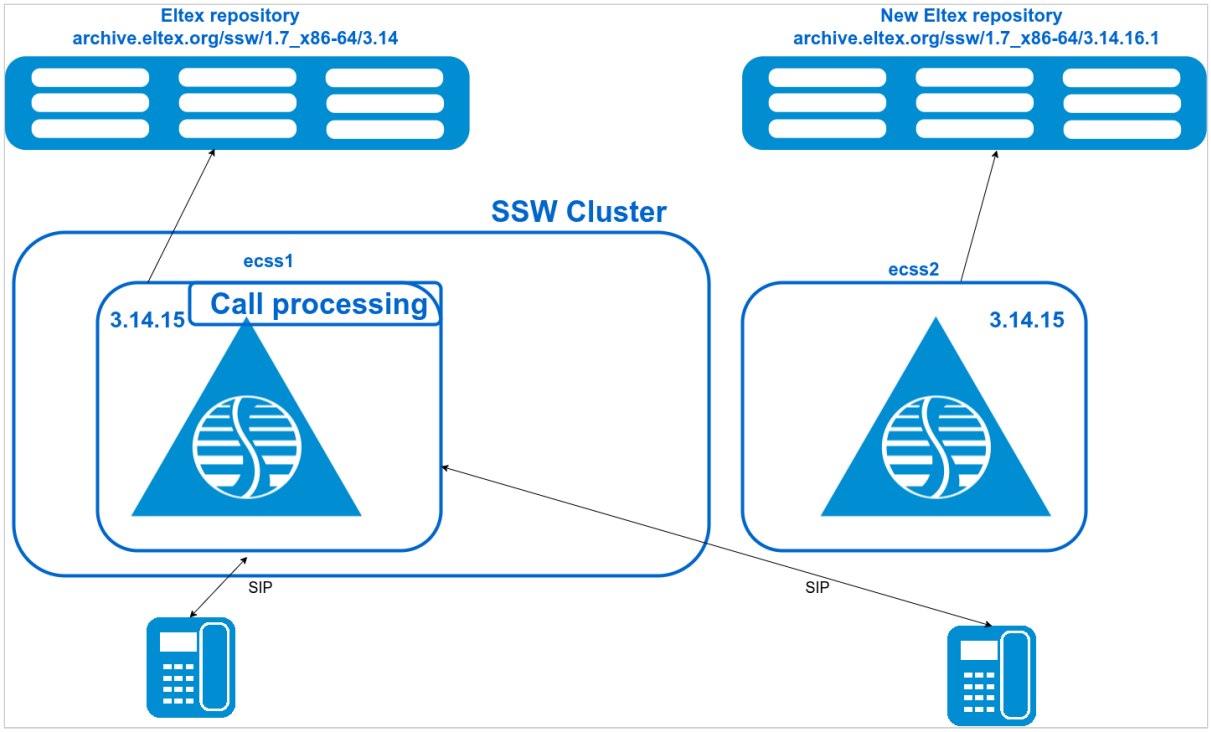
SSW 3.14.15 до обновления



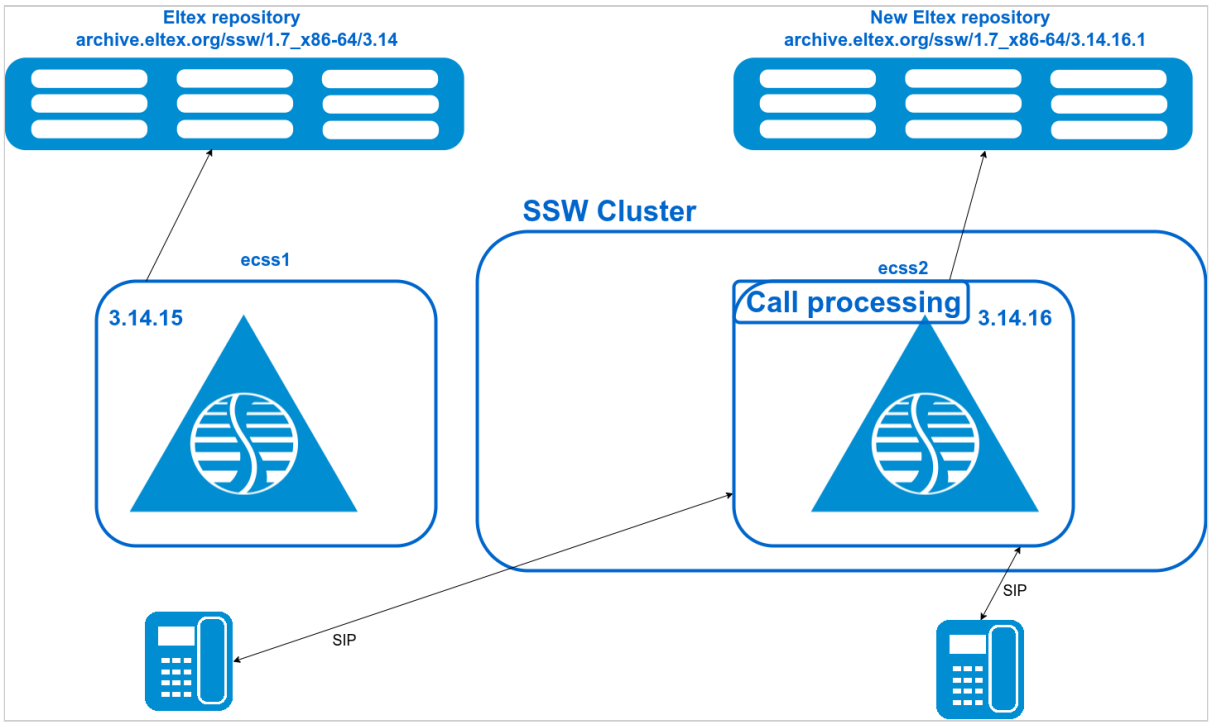
Шаг первый - "подготовительные действия на ecss2"



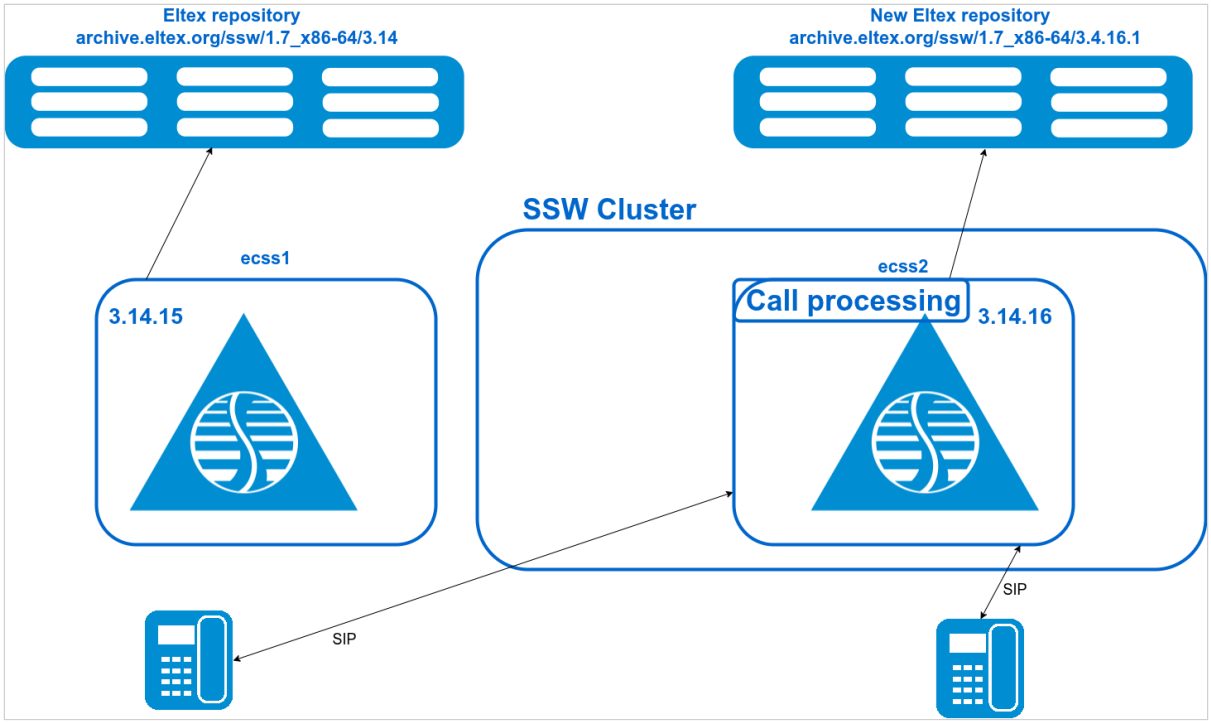
Шаг второй - подготовка к обновлению сервисов SSW на ecss2



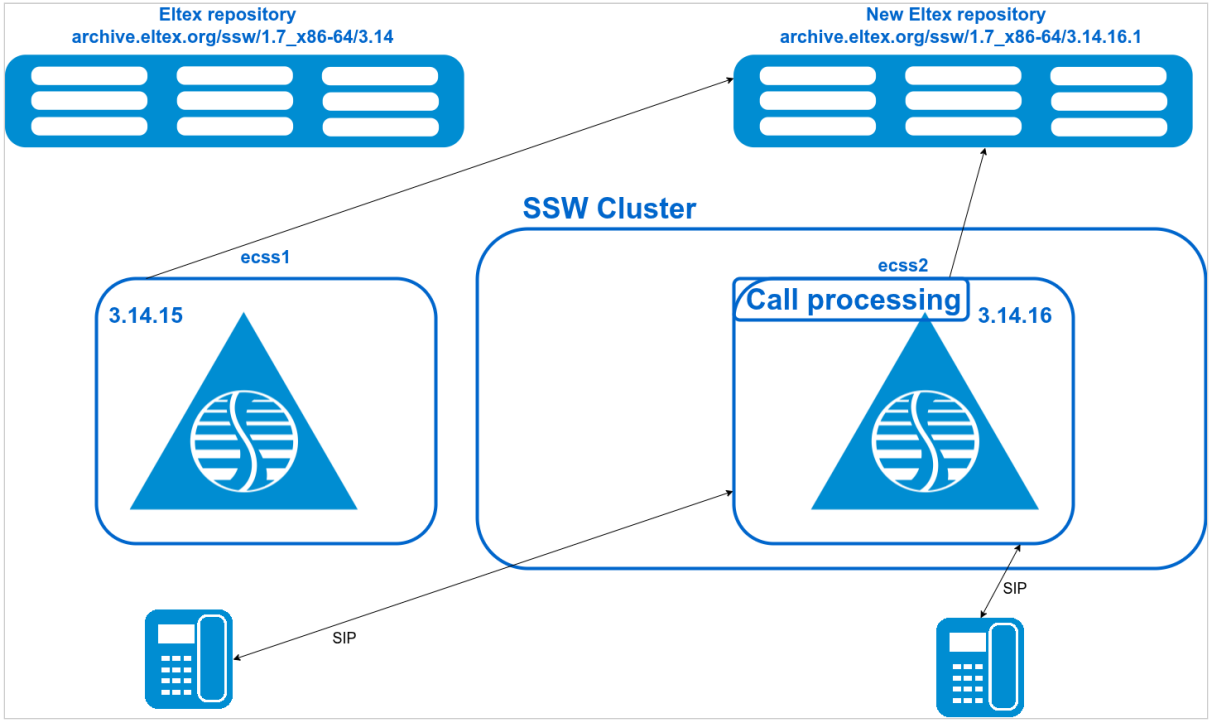
Шаг третий - обновление сервисов SSW на ecss2



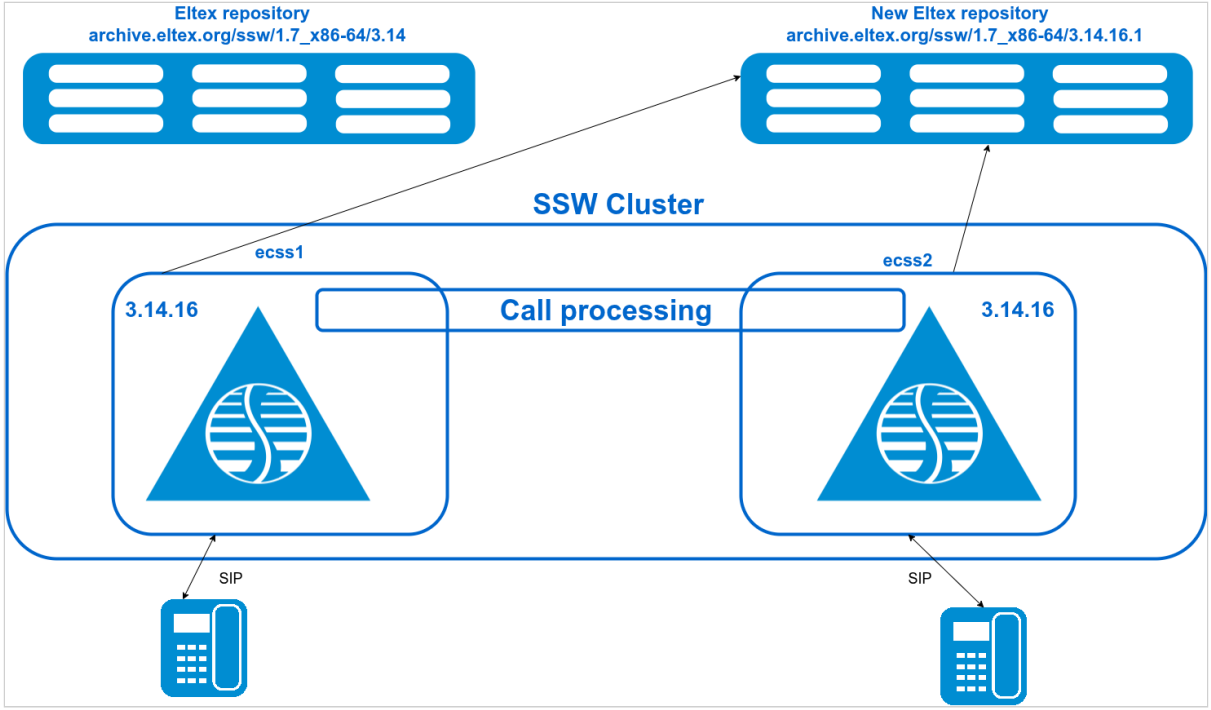
Шаг четвертый - подготовительные действия на ecss1



Шаг пятый - подготовка к обновлению сервисов SSW на ecss1



Шаг шестой - обновление сервисов SSW на ecss1



1 Подготовительные действия на ecss2

a. Выполнить полный «бэкап» (см. «Регламентное обслуживание и резервное копирование»).
b. Если нужны последние данные биллинга, то необходимо финализировать текущую cdr и скачать CDR файлы на удаленный сервер:

```
/domain/test_domain/cdr/make_finalize_cdr
Created file(s): cdr_20250822_11_51_16_1_c.csv
```

c. Включить настройку для автоматической синхронизации файлов, находящихся в сплите (по большему размеру). Подробнее см. "Проблемы связанные с возникновением split-brain" .

на ecss1

```
sudo gluster volume set ecss_volume cluster.favorite-child-policy size
```

d. При обновлении кластера следует обновлять сервера по очереди. Таким образом, при обновлении один из серверов будет всегда запущен и работа сервисов не будет прервана. Для разделения кластера следует использовать инструмент ecss-control на обоих серверах (Описание команд утилиты приведено в «Приложении Н»).
Обмен sip сигнализацией происходит через SIP-VRRP IP адрес. Аналогичным образом, через MySQL-VRRP IP адрес, происходит подключение к MySQL. Этот функционал реализован для возможности миграции Ip адреса на другой хост в случае падения/отключения/ техобслуживания основного хоста. Контроль миграции IP адресов осуществляется сервисом keepalive на каждом хосте. Для перевода трафика на первый сервер (**ecss1**), необходимо выключить keepalive на втором хосте (**ecss2**). В этом случае оба SIP-VRRP IP адреса, а так же MySQL-VRRP IP адрес, будут находится на первом хосте (**ecss1**). Выполнить следующую команду:

на ecss2

```
sudo systemctl disable --now keepalived.service
```

e. Командой **ifconfig** проверить, что **все VRRP IP-адреса** находятся на **ecss1 (VRRP-mysql/ VRRP-SIP1/VRRP-SIP2)**.

на ecss1

```
ifconfig
```

f. На **ecss1** выполнить разделение SSW на два независимых сервера следующей командой:

Команда **ecss-control** работает со стандартными именами хостов : **ecss1**/**ecss2**.

на ecss1

```
sudo ecss-control detach
```

Проверить статус:

на ecss1

```
sudo ecss-control status
```

Результат:

```
sudo ecss-control detach
detach host: ecss2, ip: 10.0.10.82
complete
sudo ecss-control status
locked ip: 10.0.10.82
```

⚠ Если после разделения на первом хосте при попытке вызова/регистрации появляется ошибка "**503 Service Unavailable**". То на проблемном хосте выполните следующие команды: **ecss-attach ecss-pa-sip**

```
ecss-attach ecss-pa-sip
**********************************************************************
************
! You attached to systemd-service ecss-pa-sip:
sip1@ecss1.                          !
! To exit, press Ctrl+C,
Ctrl+C.                                                          !
**********************************************************************
************
Erlang/OTP 21 [erts-10.3.5.10] [source] [64-bit] [smp:8:8]
[ds:8:8:10] [async-threads:1] [hipe]

Eshell V10.3.5.10  (abort with ^G)
(sip1@ecss1)1> sip_config:tring_is_connected().
false
(sip1@ecss1)2> sip_config:tring_is_connected(true).
ok
(sip1@ecss1)3> sip_config:tring_is_connected().
true
```

g. После разделения кластера необходимо замаскировать все ноды на сервере, который будет обновляться первым:

на ecss2

```
sudo systemctl mask --now ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip
ecss-sorm
```

✖ Нода **sorm** идет одна на кластер. В случае её работы на **ecss1**, при выполнении приложенной команды будет уведомление.

```
Unit /etc/systemd/system/ecss-sorm.service is masked, ignoring.
```

Данное сообщение не является ошибкой. Продолжать выполнение процедуры.

2 Подготовка к обновлению сервисов SSW на ecss2

a. Выполнить сплит на **ecss2**:

✖ Команда **ecss-control** работает со стандартными именами хостов : **ecss1**/**ecss2**.

на ecss2

```
sudo ecss-control detach
```

b. Актуализировать репозиторий ssw:

на ecss2

```
sudo sh -c "echo deb [arch=amd64] http://archive.eltex.org/ssw/1.7_x86-64/3.14.16.1
stable main extras external > /etc/apt/sources.list.d/eltex-ecss10-stable.list"
```

c. Временно зафиксировать версию пакетов ecss-*, на которые надо будет обновиться:

на ecss2

```
sudo sh -c "echo 'Package: ecss-*\nPin: version 14.16.1.*\nPin-Priority:
1001\n\nPackage: erlang*\nPin: version 2:24.3.10*\nPin-Priority: 1001' >> /etc/apt/
preferences.d/eltex-ecss10"
```

d. Выполнить обновление пакетов:

на ecss2

```
sudo apt update
```

3 Обновление сервисов SSW на ecss2

a. Обновление пакета ecss-dns-env:

на ecss2

```
sudo apt install ecss-dns-env
```

При обновлении нужно применить **новые** конфигурационные файлы (подтверждением -Y)

После обновления, нужно актуализировать конфигурацию, выполнив команду. Ответы на вопросы аналогичны процедуре инсталляции:

на ecss2

```
sudo dpkg-reconfigure ecss-dns-env
```

b. Выполнить обновление пакета ecss-user:

на ecss2

```
sudo apt install ecss-user
```

c. Обновление пакета ecss-mysql:
Установить пакет ecss-mysql:

на ecss2

```
sudo apt install ecss-mysql
```

При установке на вопрос "Path for external data-directory, where will be created tables of databases" надо указать **внешнюю директорию** хранения таблиц баз данных, либо принять значение по умолчанию.

ответы на вопросы конфигуратора
Проверьте и при необходимости актуализируйте параметры в "/etc/ecss/ecss-mysql/mysql-checker.conf".

Переменная "MASTER_HOST_IP" - это VRRP-адрес для MySQL, определенный в конфигурации keepalived .

d. Включите ecss-mysql-checker.timer:

на ecss2

```
sudo systemctl enable --now ecss-mysql-checker.timer
```

e. Обновление **Erlang** и **SSW**.
Перед обновлением включить сервисы ssw следующими командами:

на ecss2

```
sudo systemctl unmask ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip
ecss-sorm
```

на ecss2

```
sudo systemctl enable ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip
ecss-sorm
```

Перед обновлением, удалить устаревшую версию **Erlang**. Команда ниже обновит Erlang до версии 24.3, а также обновит до новейшей версии пакеты **ecss-node** и **ecss-utils**:

на ecss2

```
sudo apt remove erlang-base-hipe
```

Если **Erlang** не обновился, удалением старого пакета, то сначала обновить ecss-node и ecss-utils, а затем обновить весь **Erlang**:

на ecss2

```
sudo apt install ecss-node ecss-utils
```

на ecss2

```
sudo apt install erlang
```

После обновления **Erlang** демаскировать сервис, который был маскирован при удалении erlang-base-hipe:

на ecss2

```
sudo systemctl unmask epmd.socket epmd.service
```

f. Запуск SSW на **ecss2**.
запустить следующие сервисы:
ecss-mycelium

на ecss2

```
sudo systemctl start ecss-mycelium
```

ecss-ds

на ecss2

```
sudo systemctl start ecss-ds
```

ecss-core

на ecss2

```
sudo systemctl start ecss-core
```

ecss-mediator

на ecss2

```
sudo systemctl start ecss-mediator
```

Если лицензия поддерживает СОРМ3, то запустить также сервис ecss-sorm.

на ecss2

```
sudo systemctl start ecss-sorm
```

Если сервис **ecss-sorm** оказался **замаскирован**, то выполнить следующие команды:

на ecss2

```
sudo systemctl unmask ecss-sorm
sudo systemctl start ecss-sorm
```

Для корректной работы SORM3 необходимо выполнить коррекцию команды для sorm экстрактора. **Данную команду необходимо выполнить для каждого домена в SSW.**

выполняется в CoCon

```
/domain/<Имя_домена>/sorm3/set extractor_node sorm1@ecss2
```

Для примера:

```
/domain/test_domain/sorm3/info
┌──────────────────┬───────────┬──────────────┐
│     Property     │  Domain   │    Value     │
├──────────────────┼───────────┼──────────────┤
│adaptation        │test_domain│undefined     │
│enabled           │test_domain│false         │
│entrypoint_profile│test_domain│              │
│extractor_ip      │test_domain│127.0.0.1     │
│extractor_node    │test_domain│sorm1@ecss1   │
└──────────────────┴───────────┴──────────────┘
/domain/test_domain/sorm3/set extractor_node sorm1@ecss2
Property "extractor_node" successfully changed from:
sorm1@ecss1
   to
sorm1@ecss2.
/domain/test_domain/sorm3/info

┌──────────────────┬───────────┬──────────────┐
│     Property     │  Domain   │    Value     │
├──────────────────┼───────────┼──────────────┤
│adaptation        │test_domain│undefined     │
│enabled           │test_domain│false         │
│entrypoint_profile│test_domain│              │
│extractor_ip      │test_domain│127.0.0.1     │
│extractor_node    │test_domain│sorm1@ecss2   │
└──────────────────┴───────────┴──────────────┘
```

Команду **повторить для всех доменов** в SSW

g. Проверить статус сервисов, для этого выполнить в CoCon команду system-status:

выполняется в CoCon

```
/system-status
Checking...
┌─┬──────────────────┬──────────────────────────────────┬──────────────────┬──────────────
┬───────┐
│ │     Node         │          Release                 │ Erlang nodes     │Mnesia
nodes|Uptime|
├─┼──────────────────┼──────────────────────────────────┼──────────────────┼──────────────
┼───────┤
│ |core1@ecss2      |ecss-core-3.14.16.1.130           |core1@ecss2       |not running
|2m 27s|
│ |ds1@ecss2        |ecss-ds-3.14.16.1.130             |ds1@ecss2         |ds1@ecss2
|2m 32s|
│ |md1@ecss2        |ecss-mediator-3.14.16.1.130       |md1@ecss2         |md1@ecss2
|2m 16s|
│ |mycelium1@ecss2|ecss-mycelium-3.14.16.1.130        |mycelium1@ecss2|not running
|2m 39s|
│ |sorm1@ecss2      |ecss-sorm-3.14.16.1.130           |sorm1@ecss2       |not running
|1m 56s|
└─┴──────────────────┴──────────────────────────────────┴──────────────────┴──────────────
┴───────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────┬────────────┬────────────┐
│    Node      │    MSR   │    MSR   │ Cc-status  │ Cc-uptime  │
│              │          │  version │            │            │
├──────────────┼──────────┼──────────┼────────────┼────────────┤
│ core1@ecss2  │ msr.ecss1 │ 3.14.15.243 │ connected │ 00:00:10  │
│              │ msr.ecss2 │ 3.14.15.243 │ connected │ 00:00:10  │
└──────────────┴──────────┴──────────┴────────────┴────────────┘
```

h. Обновление пакета **ecss-restfs**.

Принять **все новые** конфигурационные файлы, выбирая **Y**.

на ecss2

```
sudo apt install ecss-restfs
```

i. Обновление пакета **ecss-media-server**.
Предварительно отключить регистрацию на рабочий хост **ecss1**. Для этого надо закомментировать в конфигурационном файле строку с realm рабочего хоста (в примере realm="sip:10.0.20.81:5000"), следующей командой (добавив <!- - вначале строки , и - -> в конце строки), следующей командой:

на ecss2

```
sudo nano /etc/ecss/ecss-media-server/conf.d/default.xml
```

пример

```
<?xml version="1.0"?>
<config>
  <accounts>
          <!--<dynamic msr_name="msr.ecss2" realm="sip:10.0.20.81:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>-->
          <dynamic msr_name="msr.ecss2" realm="sip:10.0.20.82:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>
  </accounts>
</config>
```

Выполнить обновление:

на ecss2

```
sudo apt install ecss-media-server
```

Актуализировать конфигурацию:

на ecss2

```
sudo dpkg-reconfigure ecss-media-server
```

Выполнить проверку, и при необходимости отредактировать файл командой :

на ecss2

```
sudo nano /etc/ecss/ecss-media-server/config.xml
```

пример

пример

```
<?xml version="1.0" encoding="utf-8"?>
<config date="16:54:35 20.02.2025">
  <general log-level="3" log-rotate="yes" max-calls="8192" max-vid-calls="8192"
 max-in-group="512" load-sensor="media" load-delta="10" calls-delta="100" spool-
dir-size="100M" log-name="msr.log" log-path="/var/log/ecss/media-server" use-
srtp="disabled" enable-ice-transport="no" ice-update="no" aggressive-ice="yes"
 stun-server="" suspicious-mode="no"/>
```

```
  <transport bind-addr="10.0.20.80" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca" user="webrtc@live.com"
 password="muazkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60" rtcp-
timeout="0" rtp-timeout="350" udp-src-check="no" cn-multiplier="3" port-start="12
000" port-range="2048" tias-in-sdp="no" thread-cnt="2" vid-enc-threads="2" vid-
dec-threads="2" video-conf-layout="evenly" keyframe-interval="1000" vid-decode-
delay="100" silence-threshold="-30" dtmf-flash-disable="no" video-dscp="0" other-
dscp="0" dummy-video-src="/usr/share/ecss-media-server/video/dummy_video.yuv"
 video-enc-width="640" video-enc-height="360" finalsilence="1000" rtcp-stat-dump="
yes" dtmf-tg-fpc-loop="10" dtmf-tg-fit="1" dtmf-tg-fot="2" dtmf-tg-volume="12288"
/>
  <codec pcma="1" pcmu="2" ilbc="0" gsm="0" g722="3" g726="0" g729="0" speex="0"
 l16="0" g7221="0" opus="0" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
  <accounts>
    <!-- <dynamic msr_name="msr.name"
            realm="sip:127.0.0.1:5000"
            dtmf_mode="rfc+inband+info"
            auth_name="user"
            auth_password="password" /> -->
  </accounts>
  <pbyte>
    <mcc bind-addr="10.0.20.82" port="5700"/>
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <metrics enable="no" msr_name="msr" port="8100"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

j. Обновление пакета **ecss-media-resources:**

на ecss2

```
sudo apt install ecss-media-resources
```

k. Обновление пакета **ecss-web-conf**

на ecss2

```
sudo apt install ecss-web-conf
```

l. **Обновление пакетов опциональных сервисов**

Обновление пакета **ecss-teleconference-ui**

на ecss2

```
sudo apt install ecss-teleconference-ui
```

Обновление пакета **ecss-crm-server**

на ecss2

```
sudo apt install ecss-crm-server
```

Обновление пакета **ecss-cc-ui**

на ecss2

```
sudo apt install ecss-cc-ui
```

Обновление пакета **ecss-call-api**

на ecss2

```
sudo apt install ecss-call-api
```

Обновление пакета **ecss-elph-api**

на ecss2

```
sudo apt install ecss-elph-api
```

m. Перед переводом keepalived на обновленный хост необходимо скопировать актуальную базу ds с ecss1 (иначе база регистраций будет пустая). Поэтому сначала делаем backup БД Mnesia на **ecss1**:

выполняется в CoCon

```
/node/ds1@ecss1/oasys/backup
```

```
admin@[mycelium1@ecss1#ECSS 000089]:/$ /node/ds1@ecss1/oasys/backup
Backup was successfully created at the path: "/var/lib/ecss/oasys/
Mnesia.ds1@ecss1_20250930081517"

[exec at: 30.09.2025 15:15:18, exec time: 1s 68ms, nodes: ds1@ecss1
v.3.14.15.24.4]
```

Останавливаем ecss-ds на ecss2:

на ecss2

```
sudo systemctl stop ecss-ds
```

Командой **scp** копируем backup с **ecss1** на **ecss2**.

на ecss2

```
scp -r ecss@ecss1:/var/lib/ecss/oasys/Mnesia.ds1@ecss1_20250930081517 /var/lib/ecss/oasys
```

Выполняем локальную копию текущей конфигурации:

на ecss2

```
sudo cp -a /var/lib/ecss/oasys/Mnesia.ds1@ecss2 /var/lib/ecss/oasys/Mnesia.ds1@ecss2-bkp
```

Заменяем локальную конфигурацию на конфигурацию backup с ecss1:

на ecss2

```
sudo mv /var/lib/ecss/oasys/Mnesia.ds1@ecss1_20250930081517 /var/lib/ecss/oasys/Mnesia.ds1@ecss2
```

на ecss2

```
sudo chown -R ssw:ssw /var/lib/ecss/oasys
```

Запустить **ecss-ds** и **ecss-pa-sip** (чтобы подгрузились актуальные регистрации абонентов):

на ecss2

```
sudo systemctl start ecss-ds
```

на ecss2

```
sudo systemctl start ecss-pa-sip
```

Проверить статус сервисов командой **/system-status** :

выполняется в CoCon

```
/system-status
Checking...
┌─┬──────────────┬──────────────────────────────┬──────────────┬─────────────┬──────┐
│ │     Node     │           Release            │ Erlang nodes │Mnesia nodes│Uptime│
├─┼──────────────┼──────────────────────────────┼──────────────┼─────────────┼──────┤
│ │core1@ecss2   │ecss-core-3.14.16.1.130       │core1@ecss2   │not running  │2m 27s│
│ │ds1@ecss2     │ecss-ds-3.14.16.1.130         │ds1@ecss2     │ds1@ecss2    │2m 32s│
│ │md1@ecss2     │ecss-mediator-3.14.16.1.130   │md1@ecss2     │md1@ecss2    │2m 16s│
```

```
| |mycelium1@ecss2|ecss-mycelium-3.14.16.1.130 |mycelium1@ecss2|not running |2m 39s|
| |sip1@ecss2     |ecss-pa-sip-3.14.16.1.130   |sip1@ecss2     |sip1@ecss2  |2m 22s|
| |sorm1@ecss2    |ecss-sorm-3.14.16.1.130     |sorm1@ecss2    |not running |1m 56s|
└─┴───────────────┴────────────────────────────┴───────────────┴────────────┴──────┘

All services are started.

  Active media resource selected list specific:

┌──────────────┬──────────┬──────────────┬─────────────┬─────────────┐
│     Node     │    MSR   │     MSR      │  Cc-status  │  Cc-uptime  │
│              │          │    version   │             │             │
├──────────────┼──────────┼──────────────┼─────────────┼─────────────┤
│ core1@ecss2  │ msr.ecss1 │ 3.14.15.243 │ connected   │ 00:00:10    │
│              │ msr.ecss2 │ 3.14.15.243 │ connected   │ 00:00:10    │
└──────────────┴──────────┴──────────────┴─────────────┴─────────────┘
```

n. Запуск сервиса **keepalived**

на ecss2

```
sudo systemctl enable --now keepalived
```

o. После обновления сервисов удалить файл фиксации версии ssw:

на ecss2

```
sudo rm /etc/apt/preferences.d/eltex-ecss10
```

4 Подготовительные действия на ecss1

a. Выполнить полный «бэкап» (см. «Регламентное обслуживание и резервное копирование»).
b. Подготовить **ecss1** к обновлению, для этого, перевести весь трафик на **ecss2** (команда выполняется на ecss1):

на ecss1

```
sudo systemctl disable --now keepalived.service
```

c. Командой **ifconfig** проверьте, что **все VRRP** адреса (**VRRP IP-mysql / VRRP IP-SIP1 / VRRP IP-SIP2**) находятся **на ecss2**.

на ecss2

```
ifconfig
```

⚠ Если после переключения на **ecss2,** при попытке вызова/регистрации появляется ошибка "503 Service Unavailable". То на проблемном хосте выполните следующие команды: **ecss-attach ecss-pa-sip**

```
ecss-attach ecss-pa-sip
**********************************************************************
************
! You attached to systemd-service ecss-pa-sip:
sip1@ecss1.                          !
! To exit, press Ctrl+C,
Ctrl+C.                                                          !
**********************************************************************
************
Erlang/OTP 21 [erts-10.3.5.10] [source] [64-bit] [smp:8:8]
[ds:8:8:10] [async-threads:1] [hipe]

Eshell V10.3.5.10  (abort with ^G)
(sip1@ecss1)1> sip_config:tring_is_connected().
false
(sip1@ecss1)2> sip_config:tring_is_connected(true).
ok
(sip1@ecss1)3> sip_config:tring_is_connected().
true
```

d. После разделения кластера необходимо выключить все ноды на сервере **ecss1**:

на ecss1

```
sudo systemctl mask --now ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip
ecss-sorm
```

5 Подготовка к обновлению сервисов SSW на ecss1

a. Перед обновлением актуализировать репозиторий ssw:

на ecss1

```
sudo sh -c "echo deb [arch=amd64] http://archive.eltex.org/ssw/1.7_x86-64/3.14.16.1
stable main extras external > /etc/apt/sources.list.d/eltex-ecss10-stable.list"
```

b. Временно зафиксировать версию пакетов ecss-*, на которые надо будет обновиться:

на ecss1

```
sudo sh -c "echo 'Package: ecss-*\nPin: version 14.16.1.*\nPin-Priority: 1001\n\nPackage: erlang*\nPin: version 2:24.3.10*\nPin-Priority: 1001' >> /etc/apt/preferences.d/eltex-ecss10"
```

c. Обновить пакеты командой:

на ecss1

```
sudo apt update
```

6 Обновление сервисов SSW на ecss1

a. Обновление пакета **ecss-dns-env**.

на ecss1

```
sudo apt install ecss-dns-env
```

При обновлении принимать новые конфигурационные файлы -Y.

После обновления актуализировать конфигурацию командой:

на ecss1

```
sudo dpkg-reconfigure ecss-dns-env
```

b. Обновление пакета **ecss-user**:

на ecss1

```
sudo apt install ecss-user
```

c. Обновление пакета **ecss-mysql**:
Установить пакет **ecss-mysql**:

на ecss1

```
sudo apt install ecss-mysql
```

ответы на вопросы конфигуратора

При установке на вопрос "Path for external data-directory, where will be created tables of databases" надо указать **внешнюю директорию** хранения таблиц баз данных, либо принять значение по умолчанию.

Проверьте и при необходимости актуализируйте параметры в "/etc/ecss/ecss-mysql/mysql-checker.conf":

Переменная "MASTER_HOST_IP" - это VRRP-адрес для MySQL, определенный в конфигурации keepalived.

d. Включить ecss-mysql-checker.timer

на ecss1

```
sudo systemctl enable --now ecss-mysql-checker.timer
```

e. Обновление Erlang и SSW.
Перед обновлением включить сервисы ssw:

на ecss1

```
sudo systemctl unmask ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip
ecss-sorm
```

на ecss1

```
sudo systemctl enable ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip
ecss-sorm
```

Для обновления удалить устаревшую версию Erlang. Команда ниже обновит Erlang до версии 24.3, а также обновит до новейшей версии пакеты **ecss-node** и **ecss-utils**:

на ecss1

```
sudo apt remove erlang-base-hipe
```

Если erlang не обновился удалением старого пакета, то сначала обновить ecss-node и ecss-utils и далее обновить весь erlang командами:

на ecss1

```
sudo apt install ecss-node ecss-utils
```

на ecss1

```
sudo apt install erlang
```

После обновления Erlang демаскировать сервис, который был маскирован при удалении erlang-base-hipe:

на ecss1

```
sudo systemctl unmask epmd.socket epmd.service
```

f. Сервис **ecss-sorm** на **ecss1** сразу замаскировать, т.к. у нас заявлена работа ecss-sorm **на одном хосте** в кластере.

на ecss1

```
sudo systemctl mask ecss-sorm
```

g. Выполнить на **ecss2** объединение в кластер:

на ecss2

```
sudo ecss-control attach
```

h. Активация сервиса **Keepalived**.

на ecss1

```
sudo systemctl enable --now keepalived
```

i. Запуск **SSW** на первом хосте. На **ecss1** запустите сервис **mycelium** и в **CoCon** проверьте, что сервис поднялся и видно ноды **mycelium** хостов ecss1/ ecss2:

на ecss1

```
sudo systemctl start ecss-mycelium
```

выполняется в CoCon

```
/system-status
Checking...
┌─┬────────────────┬──────────────────────────────────┬──────────────────────────────────
──┬──────────────┬────────┐
│ │     Node       │          Release                 │         Erlang nodes
│Mnesia nodes│Uptime │
├─┼────────────────┼──────────────────────────────────┼──────────────────────────────────
──┼──────────────┼────────┤
│ │core1@ecss2     │ecss-core-3.14.16.1.130           │core1@ecss2
│not running │3h 5m  │
│ │ds1@ecss2       │ecss-ds-3.14.16.1.130             │ds1@ecss2
│ds1@ecss2   │3h 5m  │
│ │md1@ecss2       │ecss-mediator-3.14.16.1.130       │md1@ecss2
│md1@ecss2   │3h 5m  │
│ │mycelium1@ecss1│ecss-mycelium-3.14.16.1.130
│mycelium1@ecss1,mycelium1@ecss2│not running │10m 42s│
│ │mycelium1@ecss2│ecss-mycelium-3.14.16.1.130
│mycelium1@ecss1,mycelium1@ecss2│not running │3h 5m  │
│ │sip1@ecss2      │ecss-pa-sip-3.14.16.1.130         │sip1@ecss2
│sip1@ecss2  │3h 5m  │
│ │sorm1@ecss2     │ecss-sorm-3.14.16.1.130           │sorm1@ecss2
│not running │3h 5m  │
└─┴────────────────┴──────────────────────────────────┴──────────────────────────────────
──┴──────────────┴────────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────────┬──────────────┬──────────────┬──────────────┐
│     Node     │     MSR      │     MSR      │  Cc-status   │  Cc-uptime   │
│              │              │   version    │              │              │
├──────────────┼──────────────┼──────────────┼──────────────┼──────────────┤
│ core1@ecss2  │ msr.ecss1    │ 3.14.15.243  │ connected    │ 00:21:09     │
│              │ msr.ecss2    │ 3.14.16.1.0  │ connected    │ 00:00:59     │
└──────────────┴──────────────┴──────────────┴──────────────┴──────────────┘
```

j. Перед запуском **ecss-ds** на **ecss1** необходимо сделать backup директории /var/lib/ecss/oasys и удалить ее, для того чтобы с другого хоста загрузить актуальную БД. Сделать это можно переименовав существующую директорию:

После этого ecss-ds на хосте **ecss1** загрузит БД с хоста **ecss2**.

на ecss1

```
sudo mv /var/lib/ecss/oasys /var/lib/ecss/oasys-$(date +%Y%m%d-%H_%M)
```

k. На **ecss1** запустите сервис **ds** и в **CoCon** проверить, что сервис **ds** доступен на обоих хостах ecss1/ ecss2:

на ecss1

```
sudo systemctl start ecss-ds
```

выполняется в CoCon

```
/system-status
Checking...
┌─┬──────────────────┬──────────────────────────────────┬──────────────────────────────────────────────────
──┬──────────────────────┬────────┐
│ │     Node         │          Release                 │         Erlang nodes
│   Mnesia nodes     │Uptime │
├─┼──────────────────┼──────────────────────────────────┼──────────────────────────────────────────────────
──┼──────────────────────┼────────┤
│ │core1@ecss2       │ecss-core-3.14.16.1.130           │core1@ecss2
│not running         │3h 13m │
│ │ds1@ecss1         │ecss-ds-3.14.16.1.130             │ds1@ecss1,ds1@ecss2
│ds1@ecss1,ds1@ecss2 │12s    │
│ │ds1@ecss2         │ecss-ds-3.14.16.1.130             │ds1@ecss1,ds1@ecss2
│ds1@ecss1,ds1@ecss2 │3h 13m │
│ │md1@ecss2         │ecss-mediator-3.14.16.1.130       │md1@ecss2
│md1@ecss2           │3h 13m │
│ │mycelium1@ecss1   │ecss-mycelium-3.14.16.1.130
│mycelium1@ecss1,mycelium1@ecss2 │not running          │18m 33s│
│ │mycelium1@ecss2   │ecss-mycelium-3.14.16.1.130
│mycelium1@ecss1,mycelium1@ecss2 │not running          │3h 13m │
│ │sip1@ecss2        │ecss-pa-sip-3.14.16.1.130         │sip1@ecss2
│sip1@ecss2          │3h 13m │
│ │sorm1@ecss2       │ecss-sorm-3.14.16.1.130           │sorm1@ecss2
│not running         │3h 12m │
└─┴──────────────────┴──────────────────────────────────┴──────────────────────────────────────────────────
──┴──────────────────────┴────────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────────┬──────────┬──────────┐
│    Node      │    MSR   │      MSR     │ Cc-status │ Cc-uptime │
│              │          │    version   │          │          │
├──────────────┼──────────┼──────────────┼──────────┼──────────┤
│ core1@ecss2 │ msr.ecss1 │ 3.14.15.243  │ connected │ 00:29:01 │
│              │ msr.ecss2 │ 3.14.16.1.0  │ connected │ 00:08:50 │
└──────────────┴──────────┴──────────────┴──────────┴──────────┘
```

l. Далее такими же действиями запустить сервисы: **ecss-mediator**, **ecss-pa-sip**, **ecss-core** следующими командами:

на ecss1

```
sudo systemctl start ecss-mediator
```

на ecss1

```
sudo systemctl start ecss-pa-sip
```

на ecss1

```
sudo systemctl start ecss-core
```

Проверить статус в CoCom:

Результат:

```
/system-status
Checking...
┌─┬────────────────┬──────────────────────────────────┬──────────────────────────────────
──┬──────────────────────────┬────────┐
│ │     Node       │          Release                 │         Erlang nodes
│    Mnesia nodes     │Uptime│
├─┼────────────────┼──────────────────────────────────┼──────────────────────────────────
──┼──────────────────────────┼────────┤
| |core1@ecss1    |ecss-core-3.14.16.1.130     |core1@ecss1,core1@ecss2
|not running          |3m 50s|
| |core1@ecss2    |ecss-core-3.14.16.1.130     |core1@ecss1,core1@ecss2
|not running          |3h 22m|
| |ds1@ecss1      |ecss-ds-3.14.16.1.130       |ds1@ecss1,ds1@ecss2
|ds1@ecss1,ds1@ecss2  |9m 42s|
| |ds1@ecss2      |ecss-ds-3.14.16.1.130       |ds1@ecss1,ds1@ecss2
|ds1@ecss1,ds1@ecss2  |3h 22m|
| |md1@ecss1      |ecss-mediator-3.14.16.1.130 |md1@ecss1,md1@ecss2
|md1@ecss1,md1@ecss2  |5m 22s|
| |md1@ecss2      |ecss-mediator-3.14.16.1.130 |md1@ecss1,md1@ecss2
|md1@ecss1,md1@ecss2  |3h 22m|
| |mycelium1@ecss1|ecss-mycelium-3.14.16.1.130
|mycelium1@ecss1,mycelium1@ecss2|not running          |28m 4s|
| |mycelium1@ecss2|ecss-mycelium-3.14.16.1.130
|mycelium1@ecss1,mycelium1@ecss2|not running          |3h 23m|
| |sip1@ecss1     |ecss-pa-sip-3.14.16.1.130   |sip1@ecss1,sip1@ecss2
|sip1@ecss1,sip1@ecss2|4m 40s|
| |sip1@ecss2     |ecss-pa-sip-3.14.16.1.130   |sip1@ecss1,sip1@ecss2
|sip1@ecss1,sip1@ecss2|3h 22m|
| |sorm1@ecss2    |ecss-sorm-3.14.16.1.130     |sorm1@ecss2
|not running          |3h 22m|
└─┴────────────────┴──────────────────────────────────┴──────────────────────────────────
```

```
──┴──────────────────────┴──────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────────┬──────────────┬──────────────┬──────────────┐
│     Node     │     MSR      │     MSR      │  Cc-status   │  Cc-uptime   │
│              │              │    version   │              │              │
├──────────────┼──────────────┼──────────────┼──────────────┼──────────────┤
│ core1@ecss1  │ msr.ecss1    │ 3.14.15.243  │ connected    │ 00:02:37     │
│ core1@ecss2  │ msr.ecss1    │ 3.14.15.243  │ connected    │ 00:38:31     │
│              │ msr.ecss2    │ 3.14.16.1.0  │ connected    │ 00:18:20     │
└──────────────┴──────────────┴──────────────┴──────────────┴──────────────┘
```

m. Обновление пакета **ecss-restfs**:

на ecss1

```
sudo apt install ecss-restfs
```

n. Обновление пакета **ecss-media-server:**

на ecss1

```
sudo apt install ecss-media-server
```

Актуализировать конфигурацию:

на ecss1

```
sudo dpkg-reconfigure ecss-media-server
```

Проверить и при необходимости исправить вручную файл → /etc/ecss/ecss-media-server/config.xml
пример

на ecss1

```
<?xml version="1.0" encoding="utf-8"?>
<config date="05:27:38 07.12.2024">
  <general log-level="3" log-rotate="yes" max-calls="8192" max-vid-
calls="8192" max-in-group="512" load-sensor="media" load-delta="10"
calls-delta="100" spool-dir-size="100M" log-name="msr.log" log-path="/
var/log/ecss/media-server" use-srtp="disabled" enable-ice-transport="no"
ice-update="no" aggressive-ice="yes" stun-server="" suspicious-mode="no"/
>
  <transport bind-addr="10.0.20.82" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca"
user="webrtc@live.com" password="muazkh"/>
```

```
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60"
rtcp-timeout="0" rtp-timeout="350" udp-src-check="no" cn-multiplier="3"
port-start="12000" port-range="2048" tias-in-sdp="no" thread-cnt="2" vid-
enc-threads="2" vid-dec-threads="2" video-conf-layout="evenly" keyframe-
interval="1000" vid-decode-delay="100" silence-threshold="-30" dtmf-
flash-disable="no" video-dscp="0" other-dscp="0" dummy-video-src="/usr/
share/ecss-media-server/video/dummy_video.yuv" video-enc-width="640"
video-enc-height="360" finalsilence="1000" rtcp-stat-dump="yes" dtmf-tg-
fpc-loop="10" dtmf-tg-fit="1" dtmf-tg-fot="2" dtmf-tg-volume="12288"/>
  <codec pcma="3" pcmu="2" ilbc="0" gsm="0" g722="3" g726="0" g729="1"
speex="5" l16="0" g7221="6" opus="0" h264="1" h263-1998="2" t38="1" tel-
event-pt="0"/>
  <accounts>
    <!-- <dynamic msr_name="msr.name"
            realm="sip:127.0.0.1:5000"
            dtmf_mode="rfc+inband+info"
            auth_name="user"
            auth_password="password" /> -->
  </accounts>
  <pbyte>
    <mcc bind-addr="10.0.20.82" port="5700"/>
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <metrics enable="no"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

Так же, для регистрации media-server`a с ecss2 на ecss1 надо убрать ранее добавленные комментирование из файла /etc/ecss/ecss-media-server/conf.d/default.xml на **ecss2**:

на ecss2

```
<?xml version="1.0"?>
<config>
  <accounts>
          <!--<dynamic msr_name="msr.ecss2" realm="sip:10.0.20.81:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>-->
          <dynamic msr_name="msr.ecss2" realm="sip:10.0.20.82:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>
  </accounts>
</config>
```

После чего перезапустить сервис:

на ecss2

```
sudo systemctl restart ecss-media-server.service
```

o. Обновление пакета **ecss-media-resources:**

на ecss1

```
sudo apt install ecss-media-resources
```

p. Обновление пакета **ecss-web-conf**:

на ecss1

```
sudo apt install ecss-web-conf
```

q. **Обновление пакетов опциональных сервисов**

Обновление пакета **ecss-teleconference-ui**

на ecss1

```
sudo apt install ecss-teleconference-ui
```

Обновление пакета **ecss-crm-server**

на ecss1

```
sudo apt install ecss-crm-server
```

Обновление пакета **ecss-cc-ui**

на ecss1

```
sudo apt install ecss-cc-ui
```

r. После обновления сервисов удалить файл фиксации версии ssw:

на ecss1

```
sudo rm /etc/apt/preferences.d/eltex-ecss10
```

7 Проверка обновления

a. После выполнения обновления проверить работу всех сервисов SSW.

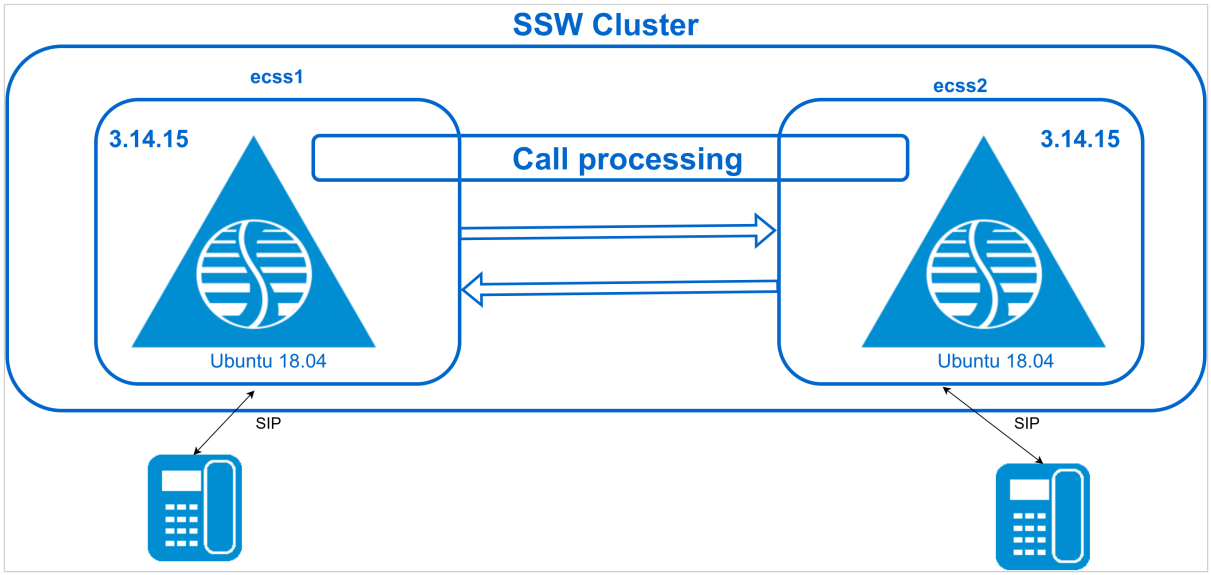
16.4 Обновление с версии 3.14.15 на 3.14.16 Ubuntu

- Обновление с версии 3.14.15 на 3.14.16
  - 1 Подготовительные действия на ecss2
  - 2 Обновление операционной системы Ubuntu 18.04 на ecss2

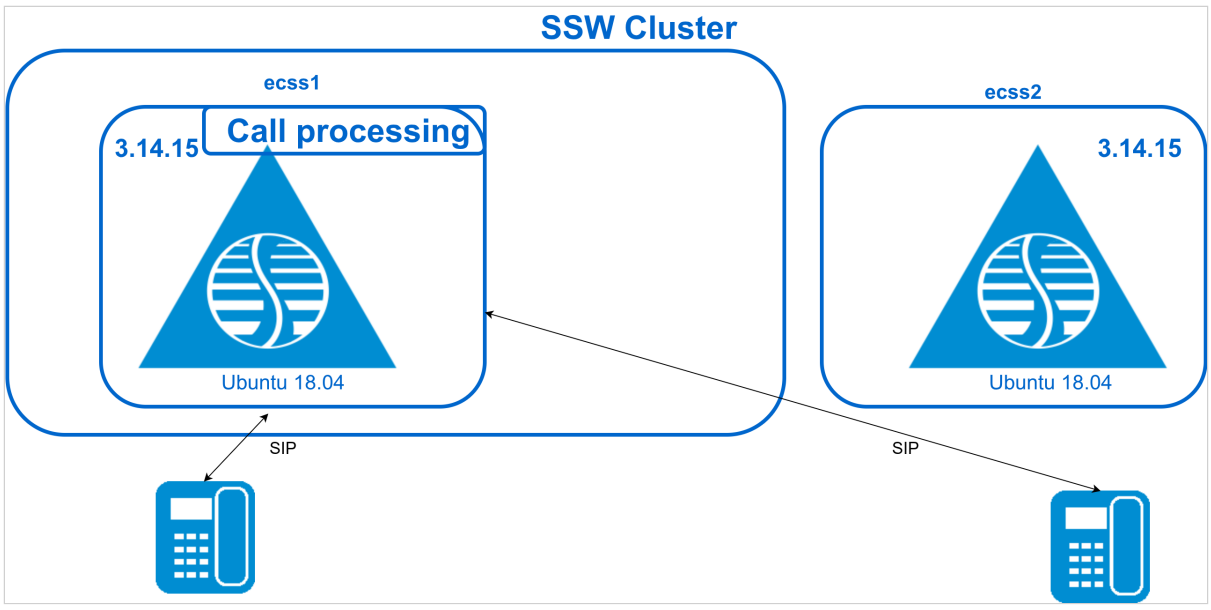
- 3 Подготовка к обновлению сервисов SSW на ecss2
- 4 Обновление сервисов SSW на ecss2
- 5 Подготовительные действия на ecss1
- 6 Обновление операционной системы Ubuntu 18.04 на ecss1
- 7 Подготовка к обновлению сервисов SSW на ecss1
- 8 Обновление сервисов SSW на ecss1
- 9 Проверка обновления

16.4.1 Обновление с версии 3.14.15 на 3.14.16

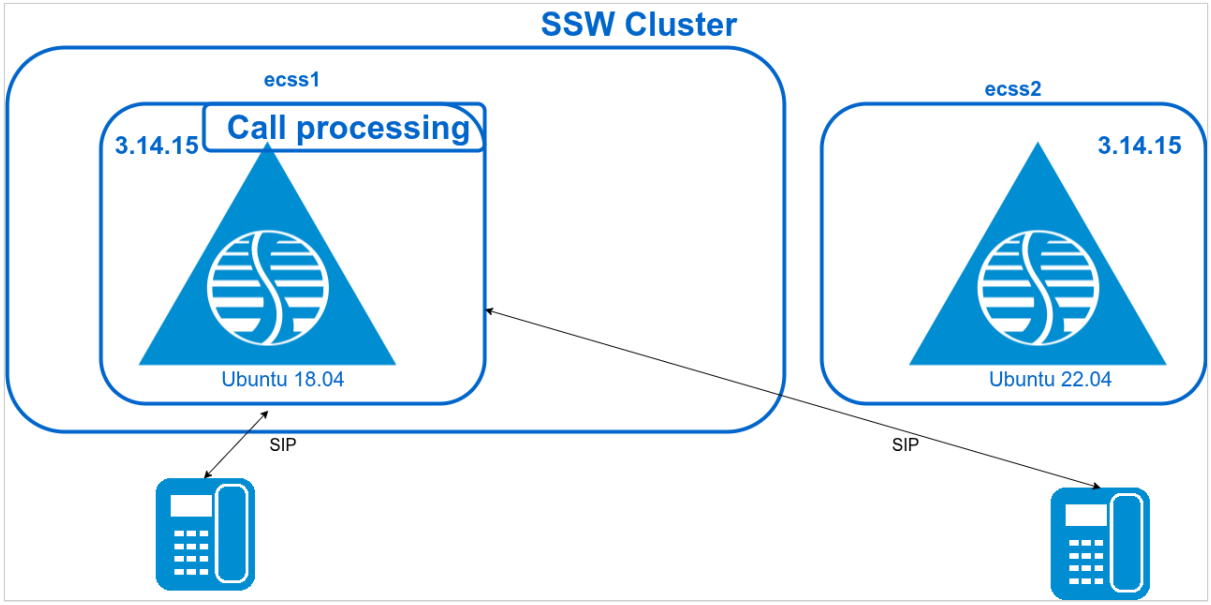
SSW 3.14.15 до обновления



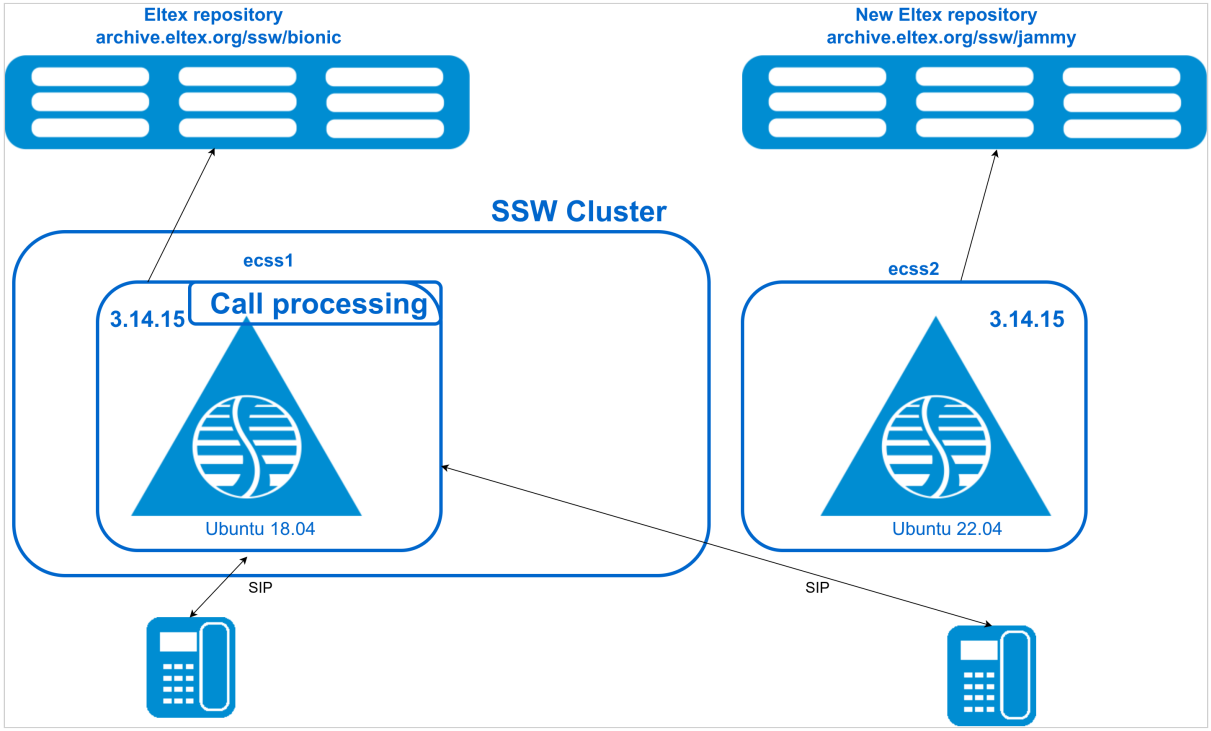
Шаг первый - "подготовительные действия на ecss2"



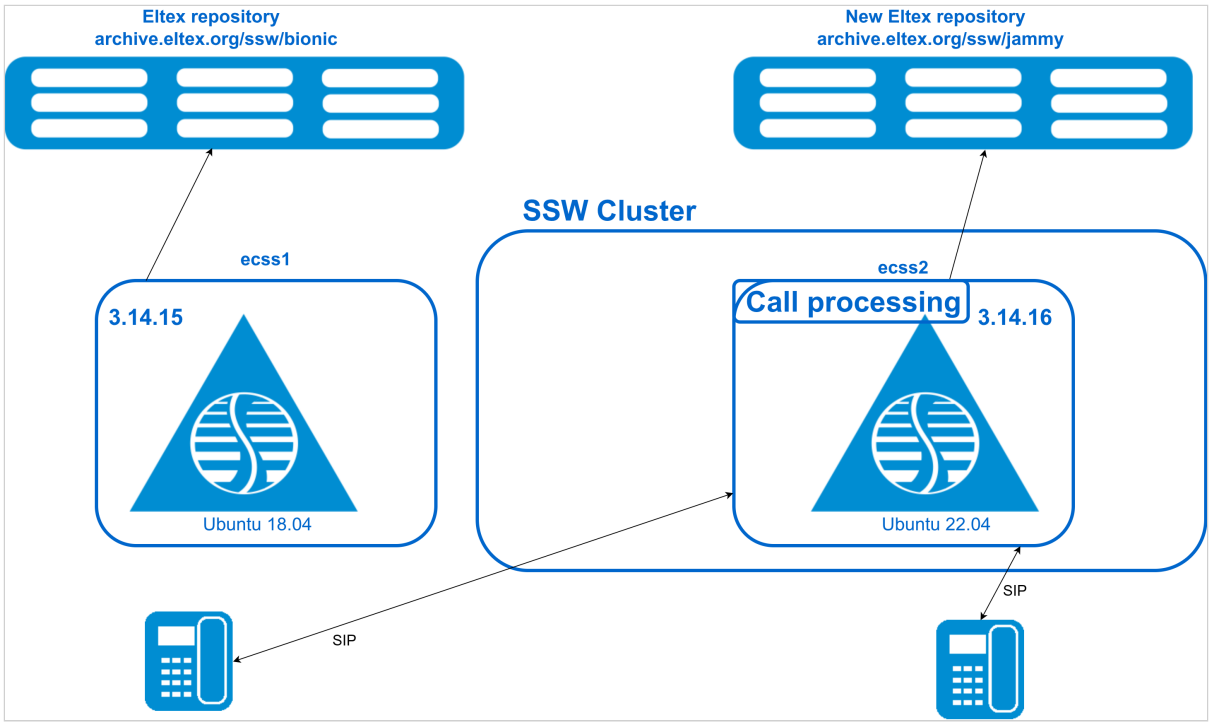
Шаг второй - обновление операционной системы Ubuntu 18.04 на ecss2



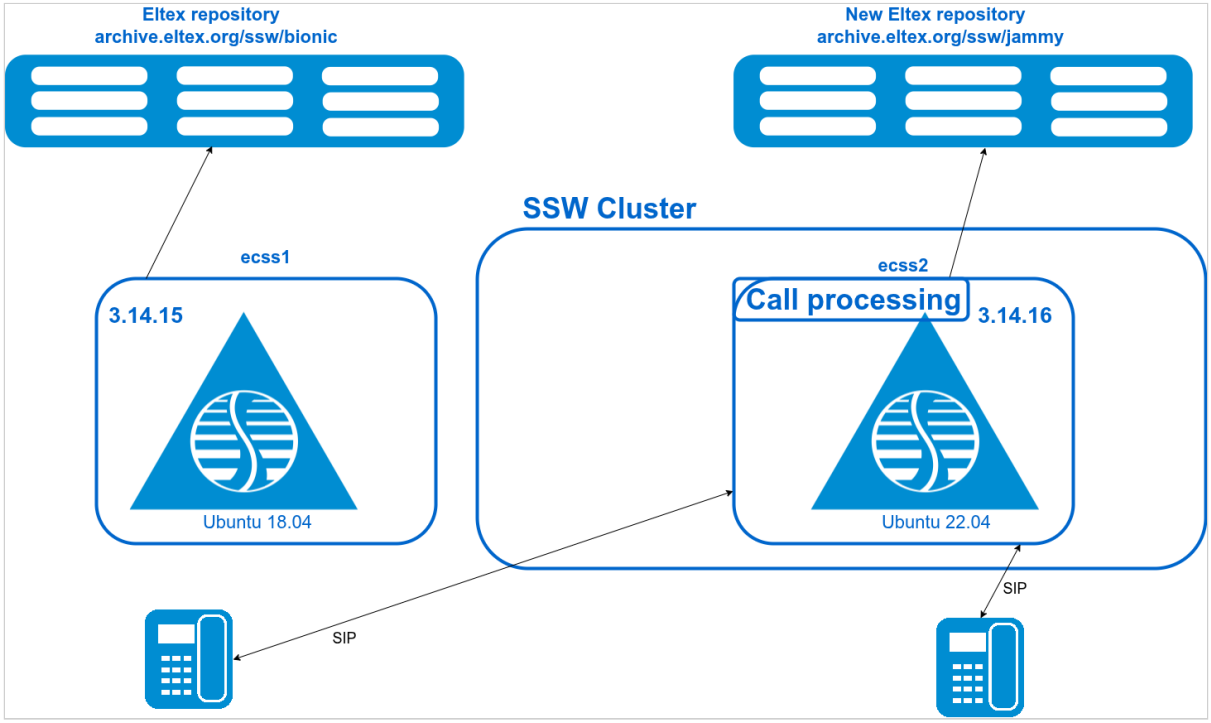
Шаг третий - подготовка к обновлению сервисов SSW на ecss2



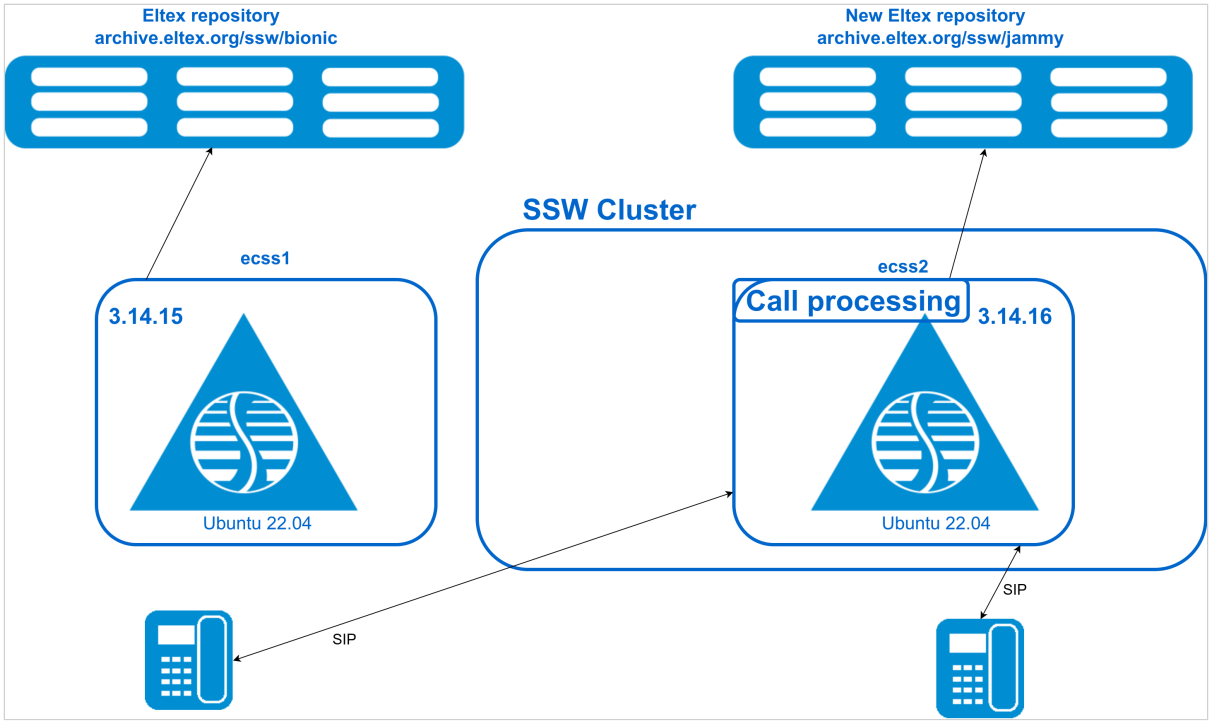
Шаг четвертый - обновление сервисов SSW на ecss2



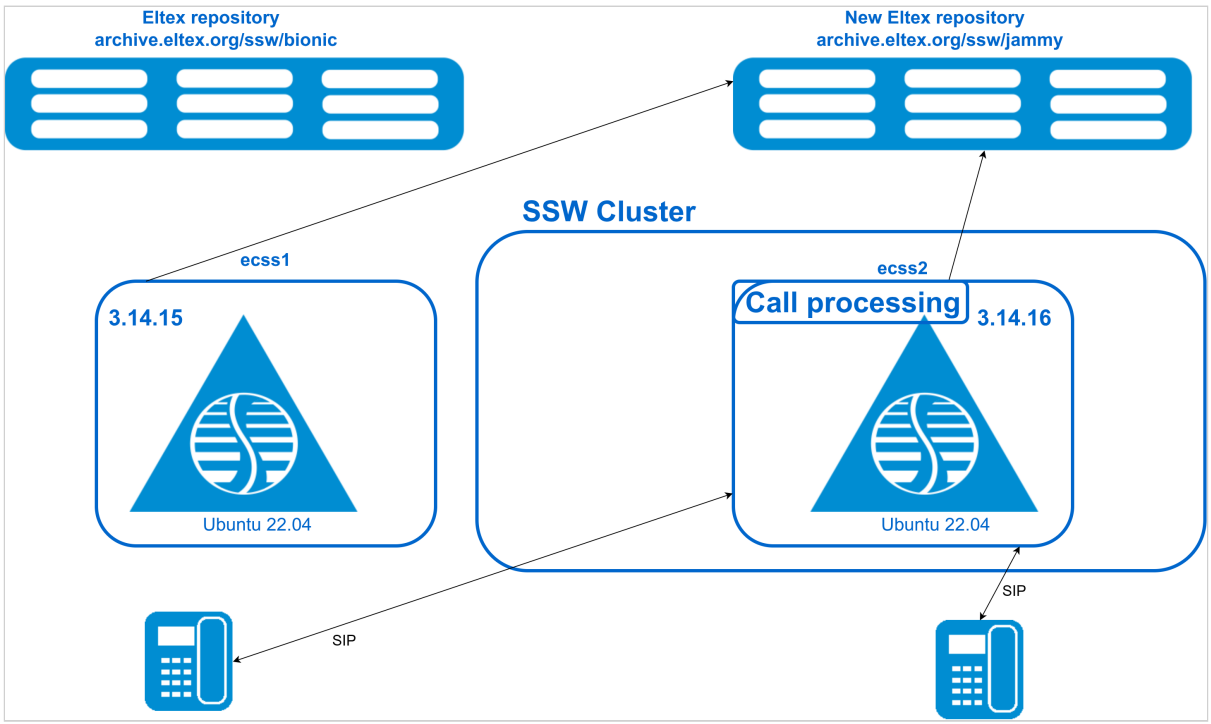
Шаг пятый - подготовительные действия на ecss1



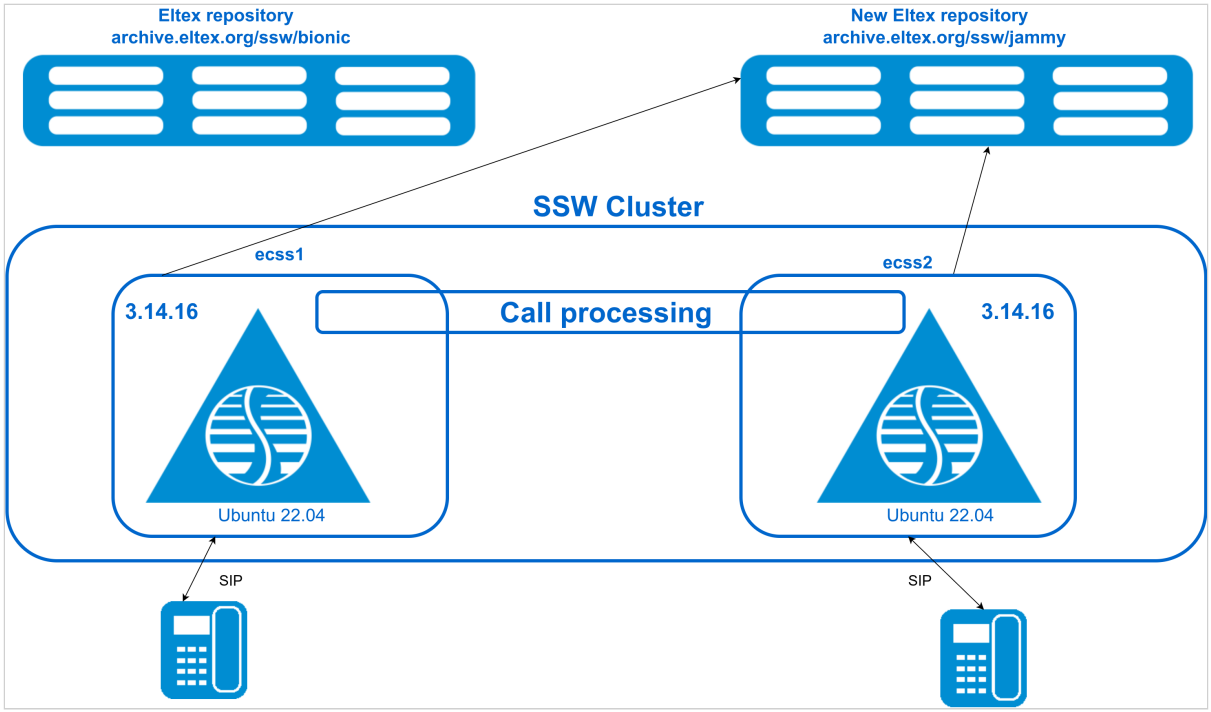
Шаг шестой - обновление операционной системы Ubuntu 18.04 на ecss1



Шаг седьмой - подготовка к обновлению сервисов SSW на ecss1



Шаг восьмой - обновление сервисов SSW на ecss1



1 Подготовительные действия на ecss2

a. Выполнить полный «бэкап» (см. «Регламентное обслуживание и резервное копирование»).
b. Если нужны последние данные биллинга, то необходимо финализировать текущую cdr и скачать CDR файлы на удаленный сервер:

```
/domain/test_domain/cdr/make_finalize_cdr
Created file(s): cdr_20250822_11_51_16_1_c.csv
```

c. Включить настройку для автоматической синхронизации файлов, находящихся в сплите (по большему размеру). Подробнее см. "Проблемы связанные с возникновением split-brain" .

на ecss1

```
sudo gluster volume set ecss_volume cluster.favorite-child-policy size
```

d. Остановить репликации MySQL.
Убедитесь, что значения параметров `Slave_IO_Running` и `Slave_SQL_Running` равны `No`.

на ecss1/ecss2

```
sudo mysql -uroot -p -e 'STOP SLAVE;'
```

на ecss1/ecss2

```
sudo mysql -uroot -p -e 'SHOW SLAVE STATUS \G;' | grep -E "Slave_IO_Running:|Slave_SQL_Running:"
```

пример

```
sudo mysql -uroot -p -e 'SHOW SLAVE STATUS \G;' | grep -E
"Slave_IO_Running:|Slave_SQL_Running:"

. . .
             Slave_IO_Running: No
            Slave_SQL_Running: No
. . .
```

e. При обновлении кластера следует обновлять сервера по очереди. Таким образом, при обновлении один из серверов будет всегда запущен и работа сервисов не будет прервана. Для разделения кластера следует использовать инструмент ecss-control на обоих серверах (Описание команд утилиты приведено в «Приложении Н»).
Обмен sip сигнализацией происходит через SIP-VRRP IP адрес. Аналогичным образом, через MySQL-VRRP IP адрес, происходит подключение к MySQL. Этот функционал реализован для возможности миграции Ip адреса на другой хост в случае падения/отключения/ техобслуживания основного хоста. Контроль миграции IP адресов осуществляется сервисом keepalive на каждом хосте. Для перевода трафика на первый сервер (**ecss1**), необходимо выключить keepalive на втором хосте (**ecss2**). В этом случае оба SIP-VRRP IP адреса, а так же MySQL-VRRP IP адрес, будут находится на первом хосте (**ecss1**). Выполнить следующую команду:

на ecss2

```
sudo systemctl disable --now keepalived.service
```

f. Командой **ifconfig** проверить, что **все VRRP IP-адреса** находятся на **ecss1 (VRRP-mysql/ VRRP-SIP1/VRRP-SIP2)**.

на ecss1

```
ifconfig
```

g. На **ecss1** выполнить разделение SSW на два независимых сервера следующей командой:

Команда **ecss-control** работает со стандартными именами хостов : **ecss1**/**ecss2**.

на ecss1

```
sudo ecss-control detach
```

Проверить статус:

на ecss1

```
sudo ecss-control status
```

Результат:

```
sudo ecss-control detach
detach host: ecss2, ip: 10.0.10.82
complete
sudo ecss-control status
locked ip: 10.0.10.82
```

⚠ Если после разделения на первом хосте при попытке вызова/регистрации появляется ошибка "**503 Service Unavailable**". То на проблемном хосте выполните следующие команды: **ecss-attach ecss-pa-sip**

```
ecss-attach ecss-pa-sip
**********************************************************************
************
! You attached to systemd-service ecss-pa-sip:
sip1@ecss1.                          !
! To exit, press Ctrl+C,
Ctrl+C.                                                          !
**********************************************************************
************
Erlang/OTP 21 [erts-10.3.5.10] [source] [64-bit] [smp:8:8]
[ds:8:8:10] [async-threads:1] [hipe]

Eshell V10.3.5.10  (abort with ^G)
(sip1@ecss1)1> sip_config:tring_is_connected().
false
(sip1@ecss1)2> sip_config:tring_is_connected(true).
ok
(sip1@ecss1)3> sip_config:tring_is_connected().
true
```

h. После разделения кластера необходимо замаскировать все ноды на сервере, который будет обновляться первым:

на ecss2

```
sudo systemctl mask --now ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip
ecss-sorm
```

⛔ Нода **sorm** идет одна на кластер. В случае её работы на **ecss1**, при выполнении приложенной команды будет уведомление.

```
Unit /etc/systemd/system/ecss-sorm.service is masked, ignoring.
```

Данное сообщение не является ошибкой. Продолжать выполнение процедуры.

i. Выполнить резервное копирование баз данных.

ВАЖНО! Корректность данных в резервной копии

Перед созданием резервной копии убедитесь, что в вашей БД таблица tolltickets не содержит устаревших полей командой:

```
sudo mysql -uroot -p -e "SHOW CREATE TABLE
ecss_calls_db.tolltickets\G;" | grep 'ss_info' || echo "Поле не
найдено"
```

Если команда показала "`ss_info` text COLLATE latin1_general_ci,", то приведите таблицу к актуальному состоянию командой:

```
sudo mysql -uroot -p -e "ALTER TABLE ecss_calls_db.tolltickets DROP
COLUMN ss_info;"
```

на ecss2

```
sudo mysqldump -uroot -p --verbose --single-transaction --quote-names --complete-
insert --extended-insert --routines --events --triggers --databases
ecss_address_book ecss_audit ecss_calls_db ecss_dialer_db ecss_meeting_db
ecss_numbers_db ecss_statistics ecss_subscribers ecss_system web_conf history_db
> ./all_dump.sql
```

j. Добавить в конфигурацию mysql параметры **skip-log-bin** и **skip-slave-start** на **ecss2** (чтобы избежать записи в бинарные логи процесса удаления БД), выполнив команду :

на ecss2

```
sudo sed -i  '40i\skip-log-bin\nskip-slave-start' /etc/mysql/mysql.conf.d/mysqld.cnf
```

k. Удалить резервный файл конфигурации (чтобы при удалении, **ecss-mysql** не загрузил конфигурацию из него):

на ecss2

```
sudo rm /var/lib/ecss/ecss-mysql/backup/mysqld.cnf
```

l. Перезапустить **ecss-mysql** с новой конфигурацией:

на ecss2

```
sudo systemctl restart mysql.service
```

m. Для корректной работы репликации после обновления необходимо сохранить конфигурационные файлы ecss-mysql, поэтому делаем отдельный backup для файлов до

обновления:

на ecss2

```
sudo cp -a /etc/ecss/ecss-mysql ~/ecss-mysql-etc-backup
```

n. Удалить **ecss-web-conf** и **ecss-mysql**, вместе с конфигурационными файлами (чтобы не было ошибок конфигураций при обновлении):

В вопросах измените IP MySQL на локальный IP БД

на ecss2

```
sudo dpkg-reconfigure ecss-web-conf
```

на ecss2

```
sudo apt purge ecss-web-conf
```

На вопросы об удалении БД → положительно отвечаем на вопросы - удалить все базы данных

на ecss2

```
sudo apt purge ecss-mysql
```

o. Проверьте, что БД удалены, выполнив команду, в выводе не должно быть других БД кроме системных ('information_schema','mysql','performance_schema','sys'):

на ecss2

```
sudo mysql -uroot -p -e "SHOW DATABASES;"
```

p. Если используется **ecss-call-api** и **ecss-elph-api**, то их нужно удалить (чтобы не было конфликтов зависимостей при обновлении).

на ecss2

```
sudo apt-get remove ecss-call-api
sudo apt-get remove ecss-elph-api
```

q. Удалить **skip-log-bin** из файла конфигурации:

на ecss2

```
sudo sed -i '/skip-log-bin/d' /etc/mysql/mysql.conf.d/mysqld.cnf
```

r. Зафиксировать пакеты mysql (чтобы пакетный менеджер при обновлении системы не удалил mysql):

на ecss2

```
sudo apt-mark hold mysql-*
```

s. Установить версию пакета nodejs (чтобы корректно было выполнено обновление дистрибутива):

на ecss2

```
sudo apt install nodejs=8.10.*
```

2 Обновление операционной системы Ubuntu 18.04 на ecss2

a. Обновление с **Ubuntu 18.04** (Bionic) до **Ubuntu 20.04** (Focal)

на ecss2

```
do-release-upgrade -m server
```

на ecss2

В случае получения сообщения:

> Checking for a new Ubuntu release
> Please install all available updates for your release before upgrading.

выполните следующие команды:

на ecss2

```
sudo apt update
```

на ecss2

```
sudo apt upgrade
```

на ecss2

```
sudo apt dist-upgrade
```

так же может потребоваться обновить пакет **ecss-user** следующей командой:

на ecss2

```
sudo apt install ecss-user
```

и затем повторить команду **do-release-upgrade -m server**:

на ecss2

```
do-release-upgrade -m server
```

Ответы на вопросы во время обновления

Во время обновления принять все **новые конфигурации пакетов ecss**.

В процессе выполнения обновления, система может предложить выполнить перезагрузку. Нужно согласится.

b. Проверка сервиса **dnsmasq**.
Если после обновления сервис **dnsmasq не находится** в активном состоянии **:**

на ecss2

```
systemctl is-active dnsmasq
```

пример

```
systemctl is-active dnsmasq
failed
```

Выполнить следующие команды:
Удалить оставшийся конфигурационный файл:

на ecss2

```
sudo rm /etc/dnsmasq.d/lxd
```

Перезапустить сервис **dnsmasq** :

на ecss2

```
sudo systemctl restart dnsmasq
```

Проверить статус сервиса:

пример

```
systemctl is-active dnsmasq
active
```

После этих действий, сервис должен быть активным и работать без ошибок.

c. Обновление пакетов **mysql**.
Отключить фиксацию пакетов:

на ecss2

```
sudo apt-mark unhold mysql-*
```

Обновить mysql до версии 8.0:

на ecss2

```
sudo apt install mysql-server
```

> ⚠ **Ответы на вопросы пакетного менеджера**
>
> **Важно**! Во время обновления mysql необходимо **оставить старую конфигурацию** файла /etc/mysql/mysql.conf.d/mysqld.conf, чтобы не потерять выставленные флаги.

После обновления необходимо актуализировать настройки mysql для версии 8.0:

на ecss2

```
sudo sed -i -e 's/^max_allowed_packet.*/max_allowed_packet = 64M/' \
       -e 's/^thread_stack.*/thread_stack = 256K/' /etc/mysql/mysql.conf.d/
mysqld.cnf
```

d. Закончить обновление хоста следующей командой:

на ecss2

```
sudo apt full-upgrade
```

Проверить версию Ubuntu следующей командой:

на ecss2

```
 lsb_release -a
```

Результат:

```
lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 20.04.6 LTS
Release:    20.04
Codename:   focal
```

e. Обновление с **Ubuntu 20.04** (Focal) до **Ubuntu 22.04** (Jammy)

на ecss2

```
do-release-upgrade -m server
```

При сообщении системы:

```
do-release-upgrade -m server
Checking for a new Ubuntu release
You have not rebooted after updating a package which requires a
reboot. Please reboot before upgrading.
```

выполнить перезагрузку следующей командой:

на ecss2

```
sudo reboot
```

а затем повторить команду:

на ecss2

```
do-release-upgrade -m server
```

Ответы на вопросы во время обновления

Во время обновления принять все **новые конфигурации пакетов ecss**.

В процессе выполнения обновления система может предложить выполнение перезагрузки. Нужно согласится.

f. После завершения обновления выполнить команду:

на ecss2

```
sudo apt autoremove
```

Данный шаг может выполняться достаточно долго

3 Подготовка к обновлению сервисов SSW на ecss2

a. Выполнить сплит на **ecss2**:

Команда **ecss-control** работает со стандартными именами хостов : **ecss1**/**ecss2**.

на ecss2

```
sudo ecss-control detach
```

b. Актуализировать репозиторий ssw:

на ecss2

```
sudo sh -c "echo deb [arch=amd64] http://archive.eltex.org/ssw/jammy/3.14.16.1
stable main extras external > /etc/apt/sources.list.d/eltex-ecss10-stable.list"
```

c. Временно зафиксировать версию пакетов ecss-*, на которые надо будет обновиться:

на ecss2

```
sudo sh -c "echo 'Package: ecss-*\nPin: version 14.16.1.*\nPin-Priority:
1001\n\nPackage: erlang*\nPin: version 2:24.3.10*\nPin-Priority: 1001' >> /etc/apt/
preferences.d/eltex-ecss10"
```

d. Выполнить обновление пакетов:

на ecss2

```
sudo apt update
```

e. Выключаем nginx сервис:

на ecss2

```
sudo systemctl disable --now nginx
```

4 Обновление сервисов SSW на ecss2

a. Обновление пакета ecss-dns-env:

на ecss2

```
sudo apt install ecss-dns-env
```

При обновлении нужно применить **новые** конфигурационные файлы (подтверждением -Y)

После обновления, нужно актуализировать конфигурацию, выполнив команду. Ответы на вопросы аналогичны процедуре инсталляции:

на ecss2

```
sudo dpkg-reconfigure ecss-dns-env
```

b. Выполнить обновление пакета ecss-user:

на ecss2

```
sudo apt install ecss-user
```

c. Обновление пакета ecss-mysql:

⚠ Перед обновлением пакета **ecss-mysql** для сохранения совместимости с mysql v5.7, надо включить в конфигурации mysql, **по умолчанию**, устаревший плагин :

на ecss2

```
sudo sh -c "echo '[mysqld]\ndefault_authentication_plugin =
mysql_native_password' >>  /etc/mysql/conf.d/plugin.cnf"
```

Перезапустить MySQL:

на ecss2

```
sudo systemctl restart mysql
```

Установить пакет ecss-mysql:

на ecss2

```
sudo apt install ecss-mysql
```

✓ При установке на вопрос "Path for external data-directory, where will be created tables of databases" надо указать **внешнюю директорию** хранения таблиц баз данных, либо принять значение по умолчанию.

ответы на вопросы конфигуратора

Обратно переместить backup конфигурационных файлов ecss-mysql (сохраненный ранее):

на ecss2

```
sudo rm -rf /etc/ecss/ecss-mysql && sudo cp -a ~/ecss-mysql-etc-backup /etc/ecss/
ecss-mysql
```

Проверьте и при необходимости актуализируйте параметры в "/etc/ecss/ecss-mysql/mysql-checker.conf".

Переменная "MASTER_HOST_IP" - это VRRP-адрес для MySQL, определенный в конфигурации keepalived .

d. Включите ecss-mysql-checker.timer:

на ecss2

```
sudo systemctl enable --now ecss-mysql-checker.timer
```

e. Загрузка резервной копии баз данных:
Перед загрузкой резервной копии необходимо модифицировать его, **указав внешнюю директорию** хранения таблиц баз данных. Так как таблицы баз данных ecss-mysql устанавливаются в отдельной директории (/var/lib/ecss-mysql), необходимо для всех таблиц в резервной копии указать внешнюю директорию, иначе они будут создаваться в стандартной data-директории (/var/lib/mysql). Также при загрузке резервной копии отключаем запись бинарных логов.

на ecss2

```
sudo sed -E "s/(ENGINE=(MyISAM|InnoDB))/\1 DATA DIRECTORY = '\/var\/lib\/ecss-mysql'/g" all_dump.sql | (echo "SET sql_log_bin = 0;"; cat -) | sudo mysql -uroot -p
```

После загрузки резервной копии сделайте реконфигурирование пакета ecss-mysql, чтобы прошли необходимые миграции v3.14.16, которые отсутствуют в резервной копии.

на ecss2

```
sudo dpkg-reconfigure ecss-mysql
```

Перезапустить репликацию командой и убедится, что slave сервер подтянул все изменения от мастера (**дождаться, пока переменная "Seconds_Behind_Master" не станет равна 0, а значение Relay_Master_Log_File должно сойтись со значением Master_Log_File**).

на ecss2

```
sudo mysql -uroot -p -e "START SLAVE;"
```

на ecss2

```
sudo mysql -e 'SHOW SLAVE STATUS \G;' | grep -E "Seconds_Behind_Master|Relay_Master_Log_File|Master_Log_File"
```

пример

```
sudo mysql -e 'SHOW SLAVE STATUS \G;' | grep -E "Seconds_Behind_Master|
Relay_Master_Log_File|Master_Log_File"
[sudo] password for abf:
               Master_Log_File: server-mysql-bin.001152
         Relay_Master_Log_File: server-mysql-bin.001152
         Seconds_Behind_Master: 0
```

Удалить параметр **skip-slave-start** из конфигурации mysql и резервного файла:

на ecss2

```
sudo sed -i '/skip-slave-start/d' /etc/mysql/mysql.conf.d/mysqld.cnf
sudo sed -i '/skip-slave-start/d' /var/lib/ecss/ecss-mysql/backup/mysqld.cnf
```

Перезапускать mysql не обязательно - данная опция влияет только на состояние SLAVE при старте mysql, а его запустили напрямую.

f. Обновление **Erlang** и **SSW**.
Перед обновлением включить сервисы ssw следующими командами:

на ecss2

```
sudo systemctl unmask ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip
ecss-sorm
```

на ecss2

```
sudo systemctl enable ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip
ecss-sorm
```

Перед обновлением, удалить устаревшую версию **Erlang**. Команда ниже обновит Erlang до версии 24.3, а также обновит до новейшей версии пакеты **ecss-node** и **ecss-utils**:

на ecss2

```
sudo apt remove erlang-base-hipe
```

Если **Erlang** не обновился, удалением старого пакета, то сначала обновить ecss-node и ecss-utils, а затем обновить весь **Erlang**:

на ecss2

```
sudo apt install ecss-node ecss-utils
```

на ecss2

```
sudo apt install erlang
```

После обновления **Erlang** демаскировать сервис, который был маскирован при удалении erlang-base-hipe:

на ecss2

```
sudo systemctl unmask epmd.socket epmd.service
```

g. Запуск SSW на **ecss2**.
Перед запуском изменить имя кластера на **отличное** от имени кластера указанном на **ecss1**. Например, добавьте постфикс upd в имя кластера:

Требуется по причине того, что mycelium1 и mycelium2 находят друг друга и в случае разных версий - mycelium на ecss2 падает.

на ecss2

```
sudo nano /etc/ecss/ecss-mycelium/mycelium1.config
```

на ecss2

```
{mycelium_broker, [
    {cluster_name, 'test_cluster_upd'},
```

запустить следующие сервисы:
ecss-mycelium

на ecss2

```
sudo systemctl start ecss-mycelium
```

ecss-ds

на ecss2

```
sudo systemctl start ecss-ds
```

ecss-core

на ecss2

```
sudo systemctl start ecss-core
```

!!!ПОКА НЕ ЗАПУСКАЕМ PA-SIP!!!

ecss-mediator

на ecss2

```
sudo systemctl start ecss-mediator
```

Если лицензия поддерживает СОРМ3, то запустить также сервис ecss-sorm.

на ecss2

```
sudo systemctl start ecss-sorm
```

Если сервис **ecss-sorm** оказался **замаскирован**, то выполнить следующие команды:

на ecss2

```
sudo systemctl unmask ecss-sorm
sudo systemctl start ecss-sorm
```

Для корректной работы SORM3 необходимо выполнить коррекцию команды для sorm экстрактора. **Данную команду необходимо выполнить для каждого домена в SSW.**

выполняется в CoCon

```
/domain/<Имя_домена>/sorm3/set extractor_node sorm1@ecss2
```

Для примера:

```
/domain/test_domain/sorm3/info
┌──────────────────┬───────────┬──────────────┐
│     Property     │  Domain   │    Value     │
├──────────────────┼───────────┼──────────────┤
│adaptation        │test_domain│undefined     │
│enabled           │test_domain│false         │
│entrypoint_profile│test_domain│              │
│extractor_ip      │test_domain│127.0.0.1     │
│extractor_node    │test_domain│sorm1@ecss1   │
└──────────────────┴───────────┴──────────────┘
/domain/test_domain/sorm3/set extractor_node sorm1@ecss2
Property "extractor_node" successfully changed from:
sorm1@ecss1
   to
sorm1@ecss2.
/domain/test_domain/sorm3/info
┌──────────────────┬───────────┬──────────────┐
│     Property     │  Domain   │    Value     │
├──────────────────┼───────────┼──────────────┤
│adaptation        │test_domain│undefined     │
│enabled           │test_domain│false         │
│entrypoint_profile│test_domain│              │
│extractor_ip      │test_domain│127.0.0.1     │
│extractor_node    │test_domain│sorm1@ecss2   │
└──────────────────┴───────────┴──────────────┘
```

Команду **повторить для всех доменов** в SSW

h. Проверить статус сервисов, для этого выполнить в CoCon команду system-status:

выполняется в CoCon

```
/system-status
Checking...
┌─┬─────────────────┬──────────────────────────────────┬─────────────────┬──────────────
┬───────┐
│ │     Node        │          Release                 │ Erlang nodes    │Mnesia
nodes|Uptime|
├─┼─────────────────┼──────────────────────────────────┼─────────────────┼──────────────
┼───────┤
│ |core1@ecss2     |ecss-core-3.14.16.1.130           |core1@ecss2      |not running
|2m 27s|
│ |ds1@ecss2       |ecss-ds-3.14.16.1.130             |ds1@ecss2        |ds1@ecss2
|2m 32s|
│ |md1@ecss2       |ecss-mediator-3.14.16.1.130       |md1@ecss2        |md1@ecss2
|2m 16s|
│ |mycelium1@ecss2|ecss-mycelium-3.14.16.1.130        |mycelium1@ecss2|not running
|2m 39s|
│ |sorm1@ecss2     |ecss-sorm-3.14.16.1.130           |sorm1@ecss2      |not running
|1m 56s|
└─┴─────────────────┴──────────────────────────────────┴─────────────────┴──────────────
┴───────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────────┬──────────────┬──────────────┬──────────────┐
│     Node     │     MSR      │     MSR      │  Cc-status   │  Cc-uptime   │
│              │              │    version   │              │              │
├──────────────┼──────────────┼──────────────┼──────────────┼──────────────┤
│ core1@ecss2  │ msr.ecss1    │ 3.14.15.243  │ connected    │ 00:00:10     │
│              │ msr.ecss2    │ 3.14.15.243  │ connected    │ 00:00:10     │
└──────────────┴──────────────┴──────────────┴──────────────┴──────────────┘
```

i. Обновление пакета **ecss-restfs**.

Принять **все новые** конфигурационные файлы, выбирая **Y**.

на ecss2

```
sudo apt install ecss-restfs
```

j. Обновление пакета **ecss-media-server**.
Предварительно отключить регистрацию на рабочий хост **ecss1**. Для этого надо закомментировать в конфигурационном файле строку с realm рабочего хоста (в примере realm="sip:10.0.20.81:5000"), следующей командой (добавив <!- - вначале строки , и - -> в конце строки), следующей командой:

на ecss2

```
sudo nano /etc/ecss/ecss-media-server/conf.d/default.xml
```

пример

```
<?xml version="1.0"?>
<config>
  <accounts>
          <!--<dynamic msr_name="msr.ecss2" realm="sip:10.0.20.81:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>-->
          <dynamic msr_name="msr.ecss2" realm="sip:10.0.20.82:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>
  </accounts>
</config>
```

Выполнить обновление:

на ecss2

```
sudo apt install ecss-media-server
```

Актуализировать конфигурацию:

на ecss2

```
sudo dpkg-reconfigure ecss-media-server
```

Выполнить проверку, и при необходимости отредактировать файл командой :

на ecss2

```
sudo nano /etc/ecss/ecss-media-server/config.xml
```

пример

пример

```
<?xml version="1.0" encoding="utf-8"?>
<config date="16:54:35 20.02.2025">
  <general log-level="3" log-rotate="yes" max-calls="8192" max-vid-calls="8192"
 max-in-group="512" load-sensor="media" load-delta="10" calls-delta="100" spool-
dir-size="100M" log-name="msr.log" log-path="/var/log/ecss/media-server" use-
srtp="disabled" enable-ice-transport="no" ice-update="no" aggressive-ice="yes"
 stun-server="" suspicious-mode="no"/>
```

```
  <transport bind-addr="10.0.20.80" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca" user="webrtc@live.com"
 password="muazkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60" rtcp-
timeout="0" rtp-timeout="350" udp-src-check="no" cn-multiplier="3" port-start="12
000" port-range="2048" tias-in-sdp="no" thread-cnt="2" vid-enc-threads="2" vid-
dec-threads="2" video-conf-layout="evenly" keyframe-interval="1000" vid-decode-
delay="100" silence-threshold="-30" dtmf-flash-disable="no" video-dscp="0" other-
dscp="0" dummy-video-src="/usr/share/ecss-media-server/video/dummy_video.yuv"
 video-enc-width="640" video-enc-height="360" finalsilence="1000" rtcp-stat-dump="
yes" dtmf-tg-fpc-loop="10" dtmf-tg-fit="1" dtmf-tg-fot="2" dtmf-tg-volume="12288"
/>
  <codec pcma="1" pcmu="2" ilbc="0" gsm="0" g722="3" g726="0" g729="0" speex="0"
 l16="0" g7221="0" opus="0" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
  <accounts>
    <!-- <dynamic msr_name="msr.name"
            realm="sip:127.0.0.1:5000"
            dtmf_mode="rfc+inband+info"
            auth_name="user"
            auth_password="password" /> -->
  </accounts>
  <pbyte>
    <mcc bind-addr="10.0.20.82" port="5700"/>
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <metrics enable="no" msr_name="msr" port="8100"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

k. Обновление пакета **ecss-media-resources:**

на ecss2

```
sudo apt install ecss-media-resources
```

l. Обновление пакета **ecss-web-conf**

на ecss2

```
sudo apt install ecss-web-conf
```

m. **Обновление пакетов опциональных сервисов**

Обновление пакета **ecss-teleconference-ui**

на ecss2

```
sudo apt install ecss-teleconference-ui
```

Обновление пакета **ecss-crm-server**

на ecss2

```
sudo apt install ecss-crm-server
```

Обновление пакета **ecss-cc-ui**

на ecss2

```
sudo apt install ecss-cc-ui
```

Обновление пакета **ecss-call-api**

на ecss2

```
sudo apt install ecss-call-api
```

Обновление пакета **ecss-elph-api**

на ecss2

```
sudo apt install ecss-elph-api
```

n. Перед переводом keepalived на обновленный хост необходимо скопировать актуальную базу ds с ecss1 (иначе база регистраций будет пустая). Поэтому сначала делаем backup БД Mnesia на **ecss1**:

выполняется в CoCon

```
/node/ds1@ecss1/oasys/backup
```

```
/node/ds1@ecss1/oasys/backup
Backup was successfully created at the path: "/var/lib/ecss/oasys/
Mnesia.ds1@ecss1_20250930081517"
```

Останавливаем ecss-ds на ecss2:

на ecss2

```
sudo systemctl stop ecss-ds
```

Командой **scp** копируем backup с **ecss1** на **ecss2**.

на ecss2

```
scp -r ecss@ecss1:/var/lib/ecss/oasys/Mnesia.ds1@ecss1_20250930081517 /var/lib/ecss/oasys
```

Выполняем локальную копию текущей конфигурации:

на ecss2

```
sudo cp -a /var/lib/ecss/oasys/Mnesia.ds1@ecss2 /var/lib/ecss/oasys/Mnesia.ds1@ecss2-bkp
```

Заменяем локальную конфигурацию на конфигурацию backup с ecss1:

на ecss2

```
sudo mv /var/lib/ecss/oasys/Mnesia.ds1@ecss1_20250930081517 /var/lib/ecss/oasys/Mnesia.ds1@ecss2
```

на ecss2

```
sudo chown -R ssw:ssw /var/lib/ecss/oasys
```

Запустить **ecss-ds** и **ecss-pa-sip** (чтобы подгрузились актуальные регистрации абонентов):

на ecss2

```
sudo systemctl start ecss-ds
```

на ecss2

```
sudo systemctl start ecss-pa-sip
```

Проверить статус сервисов командой **/system-status** :

выполняется в CoCon

```
/system-status
Checking...
┌─┬──────────────┬──────────────────────────────┬──────────────┬─────────────┬──────┐
│ │     Node     │           Release            │ Erlang nodes │Mnesia nodes│Uptime│
├─┼──────────────┼──────────────────────────────┼──────────────┼─────────────┼──────┤
│ │core1@ecss2   │ecss-core-3.14.16.1.130       │core1@ecss2   │not running  │2m 27s│
│ │ds1@ecss2     │ecss-ds-3.14.16.1.130         │ds1@ecss2     │ds1@ecss2    │2m 32s│
│ │md1@ecss2     │ecss-mediator-3.14.16.1.130   │md1@ecss2     │md1@ecss2    │2m 16s│
```

```
| |mycelium1@ecss2|ecss-mycelium-3.14.16.1.130 |mycelium1@ecss2|not running |2m 39s|
| |sip1@ecss2     |ecss-pa-sip-3.14.16.1.130   |sip1@ecss2     |sip1@ecss2  |2m 22s|
| |sorm1@ecss2    |ecss-sorm-3.14.16.1.130     |sorm1@ecss2    |not running |1m 56s|
└─┴───────────────┴────────────────────────────┴───────────────┴────────────┴──────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────┬────────────┬────────────┐
│     Node     │    MSR   │    MSR   │ Cc-status  │ Cc-uptime  │
│              │          │  version │            │            │
├──────────────┼──────────┼──────────┼────────────┼────────────┤
│ core1@ecss2  │ msr.ecss1 │ 3.14.15.243 │ connected │ 00:00:10 │
│              │ msr.ecss2 │ 3.14.16.1.0 │ connected │ 00:00:10 │
└──────────────┴──────────┴──────────┴────────────┴────────────┘
```

o. Запуск сервиса **keepalived**

на ecss2

```
sudo systemctl enable --now keepalived
```

p. После обновления сервисов удалить файл фиксации версии ssw:

на ecss2

```
sudo rm /etc/apt/preferences.d/eltex-ecss10
```

5 Подготовительные действия на ecss1

a. Выполнить полный «бэкап» (см. «Регламентное обслуживание и резервное копирование»).
b. Остановка репликации MySQL (на **ecss2**). Убедится, что значения параметров **Slave_IO_Running** и **Slave_SQL_Running** равны **No**.

на ecss2

```
sudo mysql -uroot -p -e 'STOP SLAVE;'
```

на ecss2

```
sudo mysql -uroot -p -e 'SHOW SLAVE STATUS \G;' | grep -E "Slave_IO_Running:|
Slave_SQL_Running:"
```

пример

```
sudo mysql -uroot -p -e 'SHOW SLAVE STATUS \G;' | grep -E
"Slave_IO_Running:|Slave_SQL_Running:"

. . .
             Slave_IO_Running: No
            Slave_SQL_Running: No
. . .
```

c. Подготовить **ecss1** к обновлению, для этого, перевести весь трафик на **ecss2** (команда выполняется на ecss1):

на ecss1

```
sudo systemctl disable --now keepalived.service
```

d. Командой **ifconfig** проверьте, что **все VRRP** адреса (**VRRP IP-mysql / VRRP IP-SIP1 / VRRP IP-SIP2**) находятся **на ecss2**.

на ecss2

```
ifconfig
```

⚠ Если после переключения на **ecss2,** при попытке вызова/регистрации появляется ошибка "503 Service Unavailable". То на проблемном хосте выполните следующие команды: **ecss-attach ecss-pa-sip**

```
ecss-attach ecss-pa-sip
**********************************************************************
************
! You attached to systemd-service ecss-pa-sip:
sip1@ecss1.                          !
! To exit, press Ctrl+C,
Ctrl+C.                                                          !
**********************************************************************
************
Erlang/OTP 21 [erts-10.3.5.10] [source] [64-bit] [smp:8:8]
[ds:8:8:10] [async-threads:1] [hipe]

Eshell V10.3.5.10  (abort with ^G)
(sip1@ecss1)1> sip_config:tring_is_connected().
false
(sip1@ecss1)2> sip_config:tring_is_connected(true).
ok
(sip1@ecss1)3> sip_config:tring_is_connected().
true
```

e. После разделения кластера необходимо выключить все ноды на сервере **ecss1**:

на ecss1

```
sudo systemctl mask --now ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip
ecss-sorm
```

f. Резервное копирование баз данных (на **ecss1**):

ВАЖНО! Корректность данных в резервной копии

Перед созданием резервной копии убедитесь, что в вашей БД таблица tolltickets не содержит устаревших полей командой:

```
sudo mysql -uroot -p -e "SHOW CREATE TABLE
ecss_calls_db.tolltickets\G;" | grep 'ss_info' || echo "Поле не
найдено"
```

Если команда показала "`ss_info` text COLLATE latin1_general_ci,", то приведите таблицу к актуальному состоянию командой:

```
sudo mysql -uroot -p -e "ALTER TABLE ecss_calls_db.tolltickets DROP
COLUMN ss_info;"
```

на ecss1

```
sudo mysqldump -uroot -p --verbose --single-transaction --quote-names --complete-
insert --extended-insert --routines --events --triggers --databases
ecss_address_book ecss_audit ecss_calls_db ecss_dialer_db ecss_meeting_db
ecss_numbers_db ecss_statistics ecss_subscribers ecss_system web_conf history_db
> ./all_dump.sql
```

g. Добавить в конфигурацию mysql параметры **skip-log-bin** и **skip-slave-start** на **ecss1** (чтобы избежать записи в бинарные логи процесса удаления БД):

на ecss1

```
sudo sed -i '40i\skip-log-bin\nskip-slave-start' /etc/mysql/mysql.conf.d/mysqld.cnf
```

h. Удалить резервный файл конфигурации (чтобы при удалении, **ecss-mysql** не загрузил конфигурацию из него):

на ecss1

```
sudo rm /var/lib/ecss/ecss-mysql/backup/mysqld.cnf
```

i. Перезапустить **mysql** с новой конфигурацией:

на ecss1

```
sudo systemctl restart mysql.service
```

j. Для корректной работы репликации после обновления необходимо сохранить конфигурационные файлы ecss-mysql, поэтому создать отдельный backup для файлов до обновления:

на ecss1

```
sudo cp -a /etc/ecss/ecss-mysql ~/ecss-mysql-etc-backup
```

k. Удалить **ecss-web-conf** и **ecss-mysql** вместе с конфигурационными файлами (чтобы не было ошибок конфигураций при обновлении):

- В вопросах измените IP MySQL на локальный IP БД

на ecss1

```
sudo dpkg-reconfigure ecss-web-conf
```

на ecss1

```
sudo apt purge ecss-web-conf
```

- На вопросы об удалении БД → положительно отвечаем на вопросы - удалить все базы данных

на ecss1

```
sudo apt purge ecss-mysql
```

l. Проверьте, что БД удалены командой:

- В выводе не должно быть других БД кроме системных ('information_schema','mysql','performance_schema','sys').

на ecss1

```
sudo mysql -uroot -p -e "SHOW DATABASES;"
```

m. Если используется **ecss-call-api** и **ecss-elph-api**, то удалите (чтобы не было конфликтов зависимостей при обновлении):

на ecss1

```
sudo apt remove ecss-call-api
sudo apt remove ecss-elph-api
```

n. Удалить **skip-log-bin** из файла конфигурации:

на ecss1

```
sudo sed -i '/skip-log-bin/d' /etc/mysql/mysql.conf.d/mysqld.cnf
```

o. Зафиксировать пакеты mysql (чтобы пакетный менеджер при обновлении системы не удалил mysql):

на ecss1

```
sudo apt-mark hold mysql-*
```

p. Установить версию пакета nodejs (чтобы корректно было выполнено обновление дистрибутива):

на ecss1

```
sudo apt install nodejs=8.10.*
```

6 Обновление операционной системы Ubuntu 18.04 на ecss1

a. Обновление с **Ubuntu 18.04** (Bionic) до **Ubuntu 20.04** (Focal)

на ecss1

```
do-release-upgrade -m server
```

В случае получения сообщения:

> Checking for a new Ubuntu release
> Please install all available updates for your release before upgrading.

выполните следующие команды:

на ecss1

```
sudo apt update
```

на ecss1

```
sudo apt upgrade
```

на ecss1

```
sudo apt dist-upgrade
```

так же может потребоваться обновить пакет **ecss-user** следующей командой:

на ecss1

```
sudo apt install ecss-user
```

и затем повторить команду **do-release-upgrade -m server**:

на ecss1

```
do-release-upgrade -m server
```

Ответы на вопросы во время обновления

Во время обновления принять все **новые конфигурации пакетов ecss**.

b. **Проверка сервиса dnsmasq**

Проверка сервиса **dnsmasq**

- Если после обновления сервис **dnsmasq не находится** в активном состоянии **:**

на ecss1

```
systemctl is-active dnsmasq
```

выполнить следующие действия

пример

```
systemctl is-active dnsmasq
failed
```

Выполнить следующие команды:
Удалить оставшийся конфигурационный файл

на ecss1

```
sudo rm /etc/dnsmasq.d/lxd
```

Перезапустить сервис

на ecss1

```
sudo systemctl restart dnsmasq
```

Проверить статус сервиса:

пример

```
systemctl is-active dnsmasq
active
```

После данных шагов, сервис должен быть активным и работать без ошибок.

Обновление пакетов **mysql**.
Убрать фиксацию пакетов:

c. **на ecss1**

```
sudo apt-mark unhold mysql-*
```

Обновить mysql до версии 8.0:

на ecss1

```
sudo apt install mysql-server
```

> **Ответы на вопросы пакетного менеджера**
>
> **Важно**! Во время обновления mysql необходимо оставить старую конфигурацию файла /etc/mysql/mysql.conf.d/mysqld.conf, чтобы не потерять выставленные флаги.

После обновления необходимо актуализировать настройки mysql для версии 8.0:

на ecss1

```
sudo sed -i -e 's/^max_allowed_packet.*/max_allowed_packet = 64M/' \
       -e 's/^thread_stack.*/thread_stack = 256K/' /etc/mysql/mysql.conf.d/
mysqld.cnf
```

d. Закончить обновление хоста командой:

на ecss1

```
sudo apt full-upgrade
```

e. Обновление с **Ubuntu 20.04** (Focal) до **Ubuntu 22.04** (Jammy)

на ecss1

```
do-release-upgrade -m server
```

> **Ответы на вопросы во время обновления**
>
> Во время обновления принять все **новые конфигурации пакетов ecss**.

f. После завершения обновления выполните:

на ecss1

```
sudo apt autoremove
```

> Данный шаг может выполняться достаточно долго

7 Подготовка к обновлению сервисов SSW на ecss1

a. Перед обновлением актуализировать репозиторий ssw:

на ecss1

```
sudo sh -c "echo deb [arch=amd64] http://archive.eltex.org/ssw/jammy/3.14.16.1
stable main extras external > /etc/apt/sources.list.d/eltex-ecss10-stable.list"
```

b. Временно зафиксировать версию пакетов ecss-*, на которые надо будет обновиться:

на ecss1

```
sudo sh -c "echo 'Package: ecss-*\nPin: version 14.16.1.*\nPin-Priority: 1001\n\nPackage: erlang*\nPin: version 2:24.3.10*\nPin-Priority: 1001' >> /etc/apt/preferences.d/eltex-ecss10"
```

c. Обновить пакеты командой:

на ecss1

```
sudo apt update
```

8 Обновление сервисов SSW на ecss1

a. Обновление пакета **ecss-dns-env**.

на ecss1

```
sudo apt install ecss-dns-env
```

При обновлении принимать новые конфигурационные файлы -Y.

После обновления актуализировать конфигурацию командой:

на ecss1

```
sudo dpkg-reconfigure ecss-dns-env
```

b. Обновление пакета **ecss-user**:

на ecss1

```
sudo apt install ecss-user
```

c. Обновление пакета **ecss-mysql**:
Т.к. оба хоста обновятся до mysql v8.0, надо отключить плагин по умолчанию в конфигурации mysql и поменять плагин пользователей на **ecss2**.

на ecss2

```
sudo rm /etc/mysql/conf.d/plugin.cnf
```

на ecss2

```
sudo mysql -uroot -p -N -e "SELECT CONCAT('ALTER USER \`', user, '\`@\`', host, '\`
IDENTIFIED WITH caching_sha2_password;')
FROM mysql.user
WHERE plugin='mysql_native_password'  AND user NOT IN ('root','replica','debian-sys-
maint','mysql.session','mysql.sys');" | sudo mysql
```

Ни в коем случае **не делать рестарт MySQL сервиса!**

Далее установить пакет **ecss-mysql**:

на ecss1

```
sudo apt install ecss-mysql
```

ответы на вопросы конфигуратора

При установке на вопрос "Path for external data-directory, where will be created tables of databases" надо указать **внешнюю директорию** хранения таблиц баз данных, либо принять значение по умолчанию.

Обратно переместить backup конфигурационных файлов ecss-mysql (сохраненный ранее):

на ecss1

```
sudo rm -rf /etc/ecss/ecss-mysql && sudo cp -a ~/ecss-mysql-etc-backup /etc/ecss/
ecss-mysql
```

Проверьте и при необходимости актуализируйте параметры в "/etc/ecss/ecss-mysql/mysql-checker.conf":

Переменная "MASTER_HOST_IP" - это VRRP-адрес для MySQL, определенный в конфигурации keepalived.

d. Включить ecss-mysql-checker.timer

на ecss1

```
sudo systemctl enable --now ecss-mysql-checker.timer
```

e. Загрузка резервной копии баз данных. Перед загрузкой резервной копии необходимо модифицировать ее, **указав внешнюю директорию** хранения таблиц баз данных. Так как таблицы баз данных ecss-mysql устанавливаются в отдельной директории (/var/lib/ecss-mysql), необходимо для всех таблиц в резервной копии указать внешнюю директорию, иначе они будут создаваться в стандартной data-директории (/var/lib/mysql). Также при загрузке резервной копии отключаем запись бинарных логов.

на ecss1

```
sed -E "s/(ENGINE=(MyISAM|InnoDB))/\1 DATA DIRECTORY = '\/var\/lib\/ecss-mysql'/g"
 all_dump.sql | (echo "SET sql_log_bin = 0;"; cat -) | sudo mysql -uroot -p
```

После загрузки резервной копии сделайте реконфигурирование пакета ecss-mysql, чтобы прошли необходимые миграции v3.14.16, которые отсутствуют в резервной копии.

на ecss1

```
sudo dpkg-reconfigure ecss-mysql
```

После этого запустить репликацию командой и убедится, что slave подтянул все изменения от мастера (**Seconds_Behind_Master - должна быть равна 0, а значение Relay_Master_Log_File должно сойтись со значением Master_Log_File**).

на ecss1

```
sudo mysql -uroot -p -e "START SLAVE"
```

на ecss1

```
sudo mysql -e 'SHOW SLAVE STATUS \G;' | grep -E "Seconds_Behind_Master|
Relay_Master_Log_File|Master_Log_File"
```

пример

```
sudo mysql -e 'SHOW SLAVE STATUS \G;' | grep -E "Seconds_Behind_Master|
Relay_Master_Log_File|Master_Log_File"
              Master_Log_File: server-mysql-bin.001152
        Relay_Master_Log_File: server-mysql-bin.001152
        Seconds_Behind_Master: 0
```

Удалить параметр **skip-slave-start** из конфигурации mysql и резервного файла:

на ecss1

```
sudo sed -i '/skip-slave-start/d' /etc/mysql/mysql.conf.d/mysqld.cnf
```

на ecss1

```
sudo sed -i '/skip-slave-start/d' /var/lib/ecss/ecss-mysql/backup/mysqld.cnf
```

Перезапускать mysql не обязательно - данная опция влияет только на состояние SLAVE при старте mysql, а его запустили напрямую.

f. Восстановить репликацию master-master, запустив репликацию на ecss2 и убедится, что slave подтянул все изменения от мастера (**Seconds_Behind_Master - должна быть равна 0, а значение Relay_Master_Log_File должно сойтись со значением Master_Log_File**):

на ecss2

```
sudo mysql -uroot -p -e "START SLAVE;"
```

на ecss2

```
sudo mysql -e 'SHOW SLAVE STATUS \G;' | grep "Seconds_Behind_Master"
```

пример

```
sudo mysql -e 'SHOW SLAVE STATUS \G;' | grep -E "Seconds_Behind_Master|
Relay_Master_Log_File|Master_Log_File"
              Master_Log_File: server-mysql-bin.001152
        Relay_Master_Log_File: server-mysql-bin.001152
        Seconds_Behind_Master: 0
```

g. Обновление Erlang и SSW.
Перед обновлением включить сервисы ssw:

на ecss1

```
sudo systemctl unmask ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip
ecss-sorm
```

на ecss1

```
sudo systemctl enable ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip
ecss-sorm
```

Для обновления удалить устаревшую версию Erlang. Команда ниже обновит Erlang до версии 24.3, а также обновит до новейшей версии пакеты **ecss-node** и **ecss-utils**:

на ecss1

```
sudo apt remove erlang-base-hipe
```

Если erlang не обновился удалением старого пакета, то сначала обновить ecss-node и ecss-utils и далее обновить весь erlang командами:

на ecss1

```
sudo apt install ecss-node ecss-utils
```

на ecss1

```
sudo apt install erlang
```

После обновления Erlang демаскировать сервис, который был маскирован при удалении erlang-base-hipe:

на ecss1

```
sudo systemctl unmask epmd.socket epmd.service
```

h. Запуск **SSW** на первом хосте.
Перед запуском изменить **имя кластера** на то же, которое было установлено на **ecss2** на предыдущем шаге.

на ecss1

```
sudo nano /etc/ecss/ecss-mycelium/mycelium1.config
```

на ecss1

```
{mycelium_broker, [
    {cluster_name, 'test_cluster_upd'},
```

i. Сервис **ecss-sorm** на **ecss1** сразу замаскировать, т.к. у нас заявлена работа ecss-sorm **на одном хосте** в кластере.

на ecss1

```
sudo systemctl mask ecss-sorm
```

j. Выполнить на **ecss2** объединение в кластер:

на ecss2

```
sudo ecss-control attach
```

k. Активация сервиса **Keepalived**.

на ecss1

```
sudo systemctl enable --now keepalived
```

l. На **ecss1** запустите сервис **mycelium** и в **CoCon** проверьте, что сервис поднялся и видно ноды **mycelium** хостов ecss1/ ecss2:

на ecss1

```
sudo systemctl start ecss-mycelium
```

выполняется в CoCon

```
/system-status
Checking...
┌─┬────────────────┬──────────────────────────────────┬──────────────────────────────────
──┬──────────────┬────────┐
│ │     Node       │          Release                 │         Erlang nodes
│Mnesia nodes│Uptime │
├─┼────────────────┼──────────────────────────────────┼──────────────────────────────────
──┼──────────────┼────────┤
│ │core1@ecss2     │ecss-core-3.14.16.1.130           │core1@ecss2
│not running │3h 5m  │
│ │ds1@ecss2       │ecss-ds-3.14.16.1.130             │ds1@ecss2
│ds1@ecss2   │3h 5m  │
│ │md1@ecss2       │ecss-mediator-3.14.16.1.130       │md1@ecss2
│md1@ecss2   │3h 5m  │
│ │mycelium1@ecss1 │ecss-mycelium-3.14.16.1.130
│mycelium1@ecss1,mycelium1@ecss2 │not running │10m 42s│
│ │mycelium1@ecss2 │ecss-mycelium-3.14.16.1.130
│mycelium1@ecss1,mycelium1@ecss2 │not running │3h 5m  │
│ │sip1@ecss2      │ecss-pa-sip-3.14.16.1.130         │sip1@ecss2
│sip1@ecss2  │3h 5m  │
│ │sorm1@ecss2     │ecss-sorm-3.14.16.1.130           │sorm1@ecss2
│not running │3h 5m  │
└─┴────────────────┴──────────────────────────────────┴──────────────────────────────────
──┴──────────────┴────────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────────┬──────────────────┬──────────────┬──────────────┐
│    Node      │    MSR       │      MSR         │ Cc-status    │ Cc-uptime    │
```

```
|              |            |    version    |           |           |
|--------------|------------|---------------|-----------|-----------|
| core1@ecss2  | msr.ecss1  | 3.14.15.243   | connected | 00:21:09  |
|              | msr.ecss2  | 3.14.16.1.0   | connected | 00:00:59  |
```

m. Перед запуском **ecss-ds** на **ecss1** необходимо сделать backup директории /var/lib/ecss/oasys и удалить ее, для того чтобы с другого хоста загрузить актуальную БД. Сделать это можно переименовав существующую директорию:

> После этого ecss-ds на хосте **ecss1** загрузит БД с хоста **ecss2**.

на ecss1

```
sudo mv /var/lib/ecss/oasys /var/lib/ecss/oasys-$(date +%Y%m%d-%H_%M)
```

n. На **ecss1** запустите сервис **ds** и в **CoCon** проверить, что сервис **ds** доступен на обоих хостах ecss1/ ecss2:

на ecss1

```
sudo systemctl start ecss-ds
```

выполняется в CoCon

```
/system-status
Checking...
┌─┬──────────────────┬──────────────────────────────────┬──────────────────────────────────────────
──┬──────────────────────────┬─────────┐
| |     Node         |          Release                 |         Erlang nodes
|   Mnesia nodes     |Uptime |
├─┼──────────────────┼──────────────────────────────────┼──────────────────────────────────────────
──┼──────────────────────────┼─────────┤
| |core1@ecss2       |ecss-core-3.14.16.1.130           |core1@ecss2
|not running         |3h 13m |
| |ds1@ecss1         |ecss-ds-3.14.16.1.130             |ds1@ecss1,ds1@ecss2
|ds1@ecss1,ds1@ecss2|12s    |
| |ds1@ecss2         |ecss-ds-3.14.16.1.130             |ds1@ecss1,ds1@ecss2
|ds1@ecss1,ds1@ecss2|3h 13m |
| |md1@ecss2         |ecss-mediator-3.14.16.1.130       |md1@ecss2
|md1@ecss2           |3h 13m |
| |mycelium1@ecss1|ecss-mycelium-3.14.16.1.130
|mycelium1@ecss1,mycelium1@ecss2|not running          |18m 33s|
| |mycelium1@ecss2|ecss-mycelium-3.14.16.1.130
|mycelium1@ecss1,mycelium1@ecss2|not running          |3h 13m |
| |sip1@ecss2        |ecss-pa-sip-3.14.16.1.130         |sip1@ecss2
|sip1@ecss2          |3h 13m |
| |sorm1@ecss2       |ecss-sorm-3.14.16.1.130           |sorm1@ecss2
```

```
|not running        |3h 12m |
└──┴──────────────────┴──────────────────────────────┴──────────────────────────────
──┴──────────────────────┴────────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────────┬────────────┬────────────┐
│    Node      │    MSR   │      MSR     │ Cc-status  | Cc-uptime  |
│              │          │   version    │            │            │
├──────────────┼──────────┼──────────────┼────────────┼────────────┤
| core1@ecss2 | msr.ecss1 | 3.14.15.243  | connected | 00:29:01  |
|             | msr.ecss2 | 3.14.16.1.0  | connected | 00:08:50  |
└──────────────┴──────────┴──────────────┴────────────┴────────────┘
```

o. Далее такими же действиями запустить сервисы: **ecss-mediator**, **ecss-pa-sip**, **ecss-core** следующими командами:

на ecss1

```
sudo systemctl start ecss-mediator
```

на ecss1

```
sudo systemctl start ecss-pa-sip
```

на ecss1

```
sudo systemctl start ecss-core
```

Проверить статус в CoCom:

Результат:

```
/system-status
Checking...
┌─┬──────────────┬──────────────────────────────┬──────────────────────────────
──┴──────────────────────┴────────┐
| |     Node     |          Release             |         Erlang nodes
|    Mnesia nodes     |Uptime|
├─┼──────────────┼──────────────────────────────┼──────────────────────────────
──┴──────────────────────┴────────┤
| |core1@ecss1   |ecss-core-3.14.16.1.130       |core1@ecss1,core1@ecss2
|not running          |3m 50s|
| |core2@ecss2   |ecss-core-3.14.16.1.130       |core1@ecss1,core1@ecss2
|not running          |3h 22m|
| |ds1@ecss1     |ecss-ds-3.14.16.1.130         |ds1@ecss1,ds1@ecss2
```

```
|ds1@ecss1,ds1@ecss2  |9m 42s|
| |ds1@ecss2      |ecss-ds-3.14.16.1.130       |ds1@ecss1,ds1@ecss2
|ds1@ecss1,ds1@ecss2  |3h 22m|
| |md1@ecss1      |ecss-mediator-3.14.16.1.130 |md1@ecss1,md1@ecss2
|md1@ecss1,md1@ecss2  |5m 22s|
| |md1@ecss2      |ecss-mediator-3.14.16.1.130 |md1@ecss1,md1@ecss2
|md1@ecss1,md1@ecss2  |3h 22m|
| |mycelium1@ecss1|ecss-mycelium-3.14.16.1.130
|mycelium1@ecss1,mycelium1@ecss2|not running          |28m 4s|
| |mycelium1@ecss2|ecss-mycelium-3.14.16.1.130
|mycelium1@ecss1,mycelium1@ecss2|not running          |3h 23m|
| |sip1@ecss1     |ecss-pa-sip-3.14.16.1.130   |sip1@ecss1,sip1@ecss2
|sip1@ecss1,sip1@ecss2|4m 40s|
| |sip1@ecss2     |ecss-pa-sip-3.14.16.1.130   |sip1@ecss1,sip1@ecss2
|sip1@ecss1,sip1@ecss2|3h 22m|
| |sorm1@ecss2    |ecss-sorm-3.14.16.1.130     |sorm1@ecss2
|not running          |3h 22m|
└─┴─────────────────┴────────────────────────────┴──────────────────────────────
──┴─────────────────────────┴──────┘

All services are started.

  Active media resource selected list specific:
```

Node	MSR	MSR version	Cc-status	Cc-uptime
core1@ecss1	msr.ecss1	3.14.15.243	connected	00:02:37
core1@ecss2	msr.ecss1	3.14.15.243	connected	00:38:31
	msr.ecss2	3.14.16.1.0	connected	00:18:20

p. Обновление пакета **ecss-restfs**:

на ecss1

```
sudo apt install ecss-restfs
```

q. Обновление пакета **ecss-media-server:**

на ecss1

```
sudo apt install ecss-media-server
```

Актуализировать конфигурацию:

на ecss1

```
sudo dpkg-reconfigure ecss-media-server
```

Проверить и при необходимости исправить вручную файл → /etc/ecss/ecss-media-server/config.xml
пример

на ecss1

```
<?xml version="1.0" encoding="utf-8"?>
<config date="05:27:38 07.12.2024">
  <general log-level="3" log-rotate="yes" max-calls="8192" max-vid-
calls="8192" max-in-group="512" load-sensor="media" load-delta="10"
calls-delta="100" spool-dir-size="100M" log-name="msr.log" log-path="/
var/log/ecss/media-server" use-srtp="disabled" enable-ice-transport="no"
ice-update="no" aggressive-ice="yes" stun-server="" suspicious-mode="no"/
>
  <transport bind-addr="10.0.20.82" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca"
user="webrtc@live.com" password="muazkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60"
rtcp-timeout="0" rtp-timeout="350" udp-src-check="no" cn-multiplier="3"
port-start="12000" port-range="2048" tias-in-sdp="no" thread-cnt="2" vid-
enc-threads="2" vid-dec-threads="2" video-conf-layout="evenly" keyframe-
interval="1000" vid-decode-delay="100" silence-threshold="-30" dtmf-
flash-disable="no" video-dscp="0" other-dscp="0" dummy-video-src="/usr/
share/ecss-media-server/video/dummy_video.yuv" video-enc-width="640"
video-enc-height="360" finalsilence="1000" rtcp-stat-dump="yes" dtmf-tg-
fpc-loop="10" dtmf-tg-fit="1" dtmf-tg-fot="2" dtmf-tg-volume="12288"/>
  <codec pcma="3" pcmu="2" ilbc="0" gsm="0" g722="3" g729="1" speex="5"
l16="0" g7221="6" opus="0" h264="1" h263-1998="2" t38="1" tel-event-
pt="0"/>
  <accounts>
    <!-- <dynamic msr_name="msr.name"
            realm="sip:127.0.0.1:5000"
            dtmf_mode="rfc+inband+info"
            auth_name="user"
            auth_password="password" /> -->
  </accounts>
  <pbyte>
    <mcc bind-addr="10.0.20.82" port="5700"/>
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <metrics enable="no"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

Так же, для регистрации media-server`a с ecss2 на ecss1 **надо убрать ранее добавленные комментирование** из файла /etc/ecss/ecss-media-server/conf.d/default.xml на **ecss2**:
на ecss2

```
<?xml version="1.0"?>
<config>
  <accounts>
          <!--<dynamic msr_name="msr.ecss2" realm="sip:10.0.20.81:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>-->
          <dynamic msr_name="msr.ecss2" realm="sip:10.0.20.82:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>
  </accounts>
</config>
```

После чего перезапустить сервис:

на ecss2

```
sudo systemctl restart ecss-media-server.service
```

r. Обновление пакета **ecss-media-resources:**

на ecss1

```
sudo apt install ecss-media-resources
```

s. Обновление пакета **ecss-web-conf**:

на ecss1

```
sudo apt install ecss-web-conf
```

t. **Обновление пакетов опциональных сервисов**

Обновление пакета **ecss-teleconference-ui**

на ecss1

```
sudo apt install ecss-teleconference-ui
```

Обновление пакета **ecss-crm-server**

на ecss1

```
sudo apt install ecss-crm-server
```

Обновление пакета **ecss-cc-ui**

на ecss1

```
sudo apt install ecss-cc-ui
```

u. После обновления сервисов удалить файл фиксации версии ssw:

на ecss1

```
sudo rm /etc/apt/preferences.d/eltex-ecss10
```

9 Проверка обновления

a. После выполнения обновления проверить работу всех сервисов SSW.

16.5 Обновление с версии 3.14.16 на 3.17

- Обновление с версии 3.14.16 на 3.17
  - 1 Подготовка хостов к обновлению
  - 2 Подготовительные действия на ecss2
  - 3 Подготовка к обновлению сервисов SSW на ecss2
  - 4 Обновление сервисов SSW на ecss2
  - 5 Подготовительные действия на ecss1
  - 6 Обновление сервисов SSW на ecss1
  - 7 Проверка обновления

16.5.1 Обновление с версии 3.14.16 на 3.17

1 Подготовка хостов к обновлению

В процессе обновления с версии 3.14.16 на версию 3.17 будет заменена База Данных с MySQL на PostgreSQL. Для работы PostgreSQL требуется отдельный раздел дискового пространства - партиция, которая должна быть смонтирована в директорию → "/srv/ecss/ecss-postgres-bdr-ssw". В версии 3.14.16 такая партиция не предусматривалась, поэтому ее требуется создать. Размерность данной партиции от 100G и более.

Вариантов несколько:

- При наличии нераспределенного пространства на HDD/SSD → создать новую партицию с файловой системой ext4.
- Подключить еще один физический (в случае виртуализации виртуальный) диск. Создать на нем нужную нам партицию postgres с точкой монтирования "/srv/ecss/ecss-postgres-bdr-ssw".

Данную операцию необходимо выполнить для двух хостов поочередно (порядок не важен).

2 Подготовительные действия на ecss2

1. Если нужны последние данные биллинга, то необходимо финализировать текущую cdr и скачать CDR файлы на удаленный сервер:

```
/domain/test_domain/cdr/make_finalize_cdr
Created file(s): cdr_20260122_11_51_16_1_c.csv
```

2. Включить настройку для автоматической синхронизации файлов, находящихся в сплите (по большему размеру). Подробнее см. "Проблемы связанные с возникновением split-brain" .

на ecss1

```
sudo gluster volume set ecss_volume cluster.favorite-child-policy size
```

3. Остановить репликации MySQL.
Убедитесь, что значения параметров `Slave_IO_Running` и `Slave_SQL_Running` равны `No` .

на ecss1/ecss2

```
sudo mysql -uroot -p -e 'STOP SLAVE;'
```

на ecss1/ecss2

```
sudo mysql -uroot -p -e 'SHOW SLAVE STATUS \G;' | grep -E "Slave_IO_Running:|
Slave_SQL_Running:"
```

Пример:

```
sudo mysql -uroot -p -e 'SHOW SLAVE STATUS \G;' | grep -E "Slave_IO_Running:|
Slave_SQL_Running:"

. . .
             Slave_IO_Running: No
            Slave_SQL_Running: No
. . .
```

4. При обновлении кластера следует обновлять сервера по очереди. Таким образом, при обновлении один из серверов будет всегда запущен и работа сервисов не будет прервана. Для разделения кластера следует использовать инструмент ecss-control на обоих серверах (Описание команд утилиты приведено в «Приложении Н»).
Обмен sip сигнализацией происходит через SIP-VRRP IP адрес. Аналогичным образом, через MySQL-VRRP IP адрес, происходит подключение к MySQL. Этот функционал реализован для возможности миграции Ip адреса на другой хост в случае падения/отключения/техобслуживания основного хоста. Контроль миграции IP адресов осуществляется сервисом keepalive на каждом хосте. Для перевода трафика на первый сервер (**ecss1**), необходимо выключить keepalive на втором хосте (**ecss2**). В этом случае оба SIP-VRRP IP адреса, а так же MySQL-VRRP IP адрес, будут находится на первом хосте (**ecss1**). Выполнить следующую команду:

на ecss2

```
sudo systemctl disable --now keepalived.service
```

5. Командой **ifconfig** проверить, что **все VRRP IP-адреса** находятся на **ecss1 (VRRP-mysql/VRRP-SIP1/VRRP-SIP2)**.

на ecss1

```
ifconfig
```

6. Необходимо замаскировать все ноды на сервере, который будет обновляться первым, отключаем автозапуск нод:

на ecss2

```
sudo systemctl mask --now ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip ecss-sorm
```

7. На **ecss1** выполнить разделение SSW на два независимых сервера следующей командой:

Команда **ecss-control** работает со стандартными именами хостов : **ecss1**/**ecss2**.

на ecss1

```
sudo ecss-control detach
```

Проверить статус:

на ecss1

```
sudo ecss-control status
```

Результат:

```
sudo ecss-control detach
detach host: ecss2, ip: 10.0.10.82
complete
sudo ecss-control status
locked ip: 10.0.10.82
```

Нода **sorm** идет одна на кластер. В случае её работы на **ecss1**, при выполнении приложенной команды будет уведомление.

```
Unit /etc/systemd/system/ecss-sorm.service is masked, ignoring.
```

Данное сообщение не является ошибкой. Продолжать выполнение процедуры.

3 Подготовка к обновлению сервисов SSW на ecss2

1. Актуализировать репозиторий ssw:

на обоих хостах

```
sudo sh -c "echo deb [arch=amd64]  http://archive.eltex.org/ssw/jammy/3.17 stable main extras external > /etc/apt/sources.list.d/eltex-ecss10-stable.list"
```

2. Выполнить обновление пакетов:

на обоих хостах

```
sudo apt update
```

4 Обновление сервисов SSW на ecss2

1. Устанавливаем новую БД Postgres (сначала на ecss1, далее на ecss2)

на обоих хостах, сначала на ecss1, потом на ecss2

```
sudo apt install ecss-postgres-bdr-ssw
```

!!!Если при установки пакета, выдается:!!!
ecss-postgres-bdr-ssw : Depends: docker-compose-plugin but it is not installable --> выполнить следующие команды:

на всех хостах

```
sudo install -m 0755 -d /etc/apt/keyrings
```

на всех хостах

```
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
```

на всех хостах

```
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

на всех хостах

```
echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

на всех хостах

```
sudo apt update
```

на обоих хостах, сначала на ecss1, потом на ecss2

```
sudo apt install ecss-postgres-bdr-ssw
```

В процессе установки будут заданы вопросы:
-- Выбираем адрес для репликации из списка (на двух хостах)
-- хотим установку в кластере (на двух хостах)
-- да я мастер (на ecss1)/ нет я не мастер (на ecss2)
-- на ecss2, указываем адрес мастера, (указанный в ответе на вопрос 1 для ecss1)

2. Проверить корректность настройки репликации.
Во время установки в директорию /srv/ecss/ecss-postgres-bdr-ssw устанавливаются скрипт postgresbdr_ssw.sh.
Для проверки репликации выполнить следующие команды:

Для проверки репликации выполнить команду на обоих хостах:

```
cd /srv/ecss/ecss-postgres-bdr-ssw
```

Для проверки репликации выполнить команду на обоих хостах:

```
sudo ./postgresbdr_ssw.sh check
```

Вывод на хостах ecss1/ecss2 должен быль одинаковый.

Пример:

```
sudo ./postgresbdr_ssw.sh check
Enter master ip please
Like 10.150.150.15
10.0.10.81
Enter slave ip please
10.0.10.82
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
таблица нод на слейве по ecss_storekeeper_db
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                          conn_dsn

---------------------
+-------------------------------------------------------------------------------
-------
 7569499877813944343 | host=10.0.10.81 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.82 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 rows)

таблица коннекций на слейве по ecss_storekeeper_db
     conn_sysid      |                                          conn_dsn

---------------------
+-------------------------------------------------------------------------------
-------
 7569499877813944343 | host=10.0.10.81 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.82 port=5439 dbname=ecss_storekeeper_db
```

```
user=postgres password=postgres1
(2 rows)
```

3. Обновление пакета ecss-dns-env:

на ecss2

```
sudo apt install ecss-dns-env
```

При обновлении нужно применить **новые** конфигурационные файлы (подтверждением -Y)

После обновления, нужно актуализировать конфигурацию, выполнив команду. Ответы на вопросы аналогичны процедуре инсталляции:

на ecss2

```
sudo dpkg-reconfigure ecss-dns-env
```

4. Выполнить обновление пакета ecss-user:

на ecss2

```
sudo apt install ecss-user
```

5. Выполнить обновление пакета ecss-utils:

на ecss2

```
sudo apt install ecss-utils
```

6. Выполнить обновление пакета ecss-node:

на ecss2

```
sudo apt install ecss-node
```

ВОПРОСЫ НА УСТАНОВКУ ecss-node:
* ECSS-node ID in DNS format: <Согласно лицензии>.
* Default configuration: yes
* Global config: yes
!!!Принимаем все изменения файлов конфигурации.!!

7. Выполнить миграцию пользователей cocon(audit_database):

на ecss2

```
/usr/bin/migrate_cocon.exs
```

При выполнении скрипта миграции необходимо будет ввести следующие параметры:

```
MySQL database hostname (cocon.mysql.ecss):
MySQL database port (3306):
ecss_audit database username (audit):
ecss_audit database password (audit):
```

В скобках, указаны дефолтные значения (соответственно, вводим свои, при использовании не дефолтных).

8. Включаем ноды.
   Включить сервисы ssw следующими командами:

на ecss2

```
sudo systemctl unmask ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip ecss-sorm
```

на ecss2

```
sudo systemctl enable ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip ecss-sorm
```

9. Запуск SSW на **ecss2**. Запустить следующие сервисы:
   ecss-mycelium

на ecss2

```
sudo systemctl start ecss-mycelium
```

ecss-ds

на ecss2

```
sudo systemctl start ecss-ds
```

ecss-core

на ecss2

```
sudo systemctl start ecss-core
```

ecss-mediator

на ecss2

```
sudo systemctl start ecss-mediator
```

- Если лицензия поддерживает СОРМ3, то запустить также сервис ecss-sorm.

  на ecss2

  ```
  sudo systemctl start ecss-sorm
  ```

  Если сервис **ecss-sorm** оказался **замаскирован**, то выполнить следующие команды:

  на ecss2

  ```
  sudo systemctl unmask ecss-sorm
  sudo systemctl start ecss-sorm
  ```

Для корректной работы SORM3 необходимо выполнить коррекцию команды для sorm экстрактора. **Данную команду необходимо выполнить для каждого домена в SSW.**

выполняется в CoCon

```
/domain/<Имя_домена>/sorm3/set extractor_node sorm1@ecss2
```

Для примера:

```
/domain/test_domain/sorm3/info
```

```
┌──────────────────┬─────────────┬──────────────┐
│     Property     │  Domain     │   Value      │
├──────────────────┼─────────────┼──────────────┤
│adaptation        │test_domain  │undefined     │
│enabled           │test_domain  │false         │
│entrypoint_profile│test_domain  │              │
│extractor_ip      │test_domain  │127.0.0.1     │
│extractor_node    │test_domain  │sorm1@ecss1   │
└──────────────────┴─────────────┴──────────────┘
```

```
/domain/test_domain/sorm3/set extractor_node sorm1@ecss2
Property "extractor_node" successfully changed from:
sorm1@ecss1
   to
sorm1@ecss2.
/domain/test_domain/sorm3/info
```

```
┌──────────────────┬─────────────┬──────────────┐
│     Property     │  Domain     │   Value      │
├──────────────────┼─────────────┼──────────────┤
│adaptation        │test_domain  │undefined     │
│enabled           │test_domain  │false         │
│entrypoint_profile│test_domain  │              │
│extractor_ip      │test_domain  │127.0.0.1     │
│extractor_node    │test_domain  │sorm1@ecss2   │
└──────────────────┴─────────────┴──────────────┘
```

Команду **повторить для всех доменов** в SSW

!!!ПОКА НЕ ЗАПУСКАЕМ PA-SIP!!!

10. Проверить статус сервисов, для этого выполнить в CoCon команду system-status:

выполняется в CoCon

```
/system-status
Checking...
┌─┬──────────────────┬──────────────────────────────────┬──────────────────┬──────────────┬──────
──┐
│ │     Node         │          Release                 │ Erlang nodes     │Mnesia
nodes│Uptime│
├─┼──────────────────┼──────────────────────────────────┼──────────────────┼──────────────┼──────
──┤
│ │core1@ecss2       │ecss-core-3.17.1.0.1771           │core1@ecss2       │not running │2m
27s│
│ │ds1@ecss2         │ecss-ds-3.17.1.0.1771             │ds1@ecss2         │ds1@ecss2     │2m
32s│
│ │md1@ecss2         │ecss-mediator-3.17.1.0.1771       │md1@ecss2         │md1@ecss2     │2m
16s│
│ │mycelium1@ecss2   │ecss-mycelium-3.17.1.0.1771       │mycelium1@ecss2   │not running │2m
39s│
│ │sorm1@ecss2       │ecss-sorm-3.17.1.0.1771           │sorm1@ecss2       │not running │1m
56s│
└─┴──────────────────┴──────────────────────────────────┴──────────────────┴──────────────┴──────
──┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────┬──────────┬──────────┐
│     Node     │    MSR   │    MSR   │ Cc-status│ Cc-uptime│
│              │          │  version │          │          │
├──────────────┼──────────┼──────────┼──────────┼──────────┤
│ core1@ecss2  │ msr.ecss1│ 3.14.16.1.0 │ connected │ 00:00:10 │
│              │ msr.ecss2│ 3.14.16.1.0 │ connected │ 00:00:10 │
└──────────────┴──────────┴──────────┴──────────┴──────────┘
```

11. Выполнить миграции таблиц DB подробное описание:
Проверить, какие данные доступны для переноса, можно командой:

выполняется в CoCon

`/system/postgresql/migration/check <DATABASE> *`
где DATABASE:

- `address_book`
- `audit`
- `calls`

- `dialer`
- `history`
- `meetings`
- `subscribers`

Пример:

```
/system/postgresql/migration/check address_book * --hostname ecss1 --username
******* --password *******
┌──────────┬──────────┬──────────────────────────────────────┬──────────────
──────────┐
│    DB    │   Table  │              Status                  │
Description          │
├──────────┼──────────┼──────────────────────────────────────┼──────────────
──────────┤
|address_book|contact         |Needs migration.                    |Depends on:
`source`.          |
|            |                |Meegrated: 0 / 1785                 |
|
|            |                |                                    |
|
|-           |label           |Needs migration.                    |No
dependencies.              |
|            |                |Meegrated: 0 / 5                    |
|
|            |                |                                    |
|
|-           |phone           |Needs migration.                    |Depends on:
`contact`, `label`.|
|            |                |Meegrated: 0 / 2493                 |
|
|            |                |                                    |
|
|-           |changes         |Needs migration.                    |No
dependencies.              |
|            |                |Meegrated: 0 / 5                    |
|
|            |                |                                    |
|
|-           |email           |Needs migration.                    |Depends on:
`contact`.         |
|            |                |Meegrated: 0 / 588                  |
|
└──────────┴──────────┴──────────────────────────────────────┴──────────────
──────────┘
```

Миграцию выполнить командой:

выполняется в CoCon

```
/system/postgresql/migration/do <DATABASE> <TABLE> --hostname ecss1
```

Проверить прохождение миграции можно командой:

выполняется в CoCon

```
/system/postgresql/migration/info <DATABASE> <TABLE>
```

Пример:

```
/system/postgresql/migration/info address_book
┌────┬────────────┬──────────────────────┬──────────┬──────────────────┬──────────────────┬──────────────────┐
│ ID │     DB     │       Table          │  Status  │     Props        │    Start Time    │     End Time     │
├────┼────────────┼──────────────────────┼──────────┼──────────────────┼──────────────────┼──────────────────┤
│1   │address_book│source                │finished  │current: 2        │22.01.2026 13:14:20│22.01.2026 13:14:20│
│    │            │                      │          │overall: 2        │                  │                  │
│2   │address_book│label                 │failed    │current: 0        │22.01.2026 13:14:30│22.01.2026 13:14:30│
│    │            │                      │          │overall: 5        │                  │                  │
│3   │address_book│changes               │failed    │current: 0        │22.01.2026 13:14:49│22.01.2026 13:14:49│
│    │            │                      │          │overall: 5        │                  │                  │
│4   │address_book│contact               │finished  │current: 1785     │22.01.2026 13:14:56│22.01.2026 13:14:56│
│    │            │                      │          │overall: 1785     │                  │                  │
│5   │address_book│phone                 │finished  │current: 2493     │22.01.2026 13:15:03│22.01.2026 13:15:03│
│    │            │                      │          │overall: 2493     │                  │                  │
│6   │address_book│email                 │finished  │current: 588      │22.01.2026 13:15:07│22.01.2026 13:15:07│
│    │            │                      │          │overall: 588      │                  │                  │
└────┴────────────┴──────────────────────┴──────────┴──────────────────┴──────────────────┴──────────────────┘
```

12. Обновление пакета **ecss-restfs**.

Принять **все новые** конфигурационные файлы, выбирая **Y**.

на ecss2

```
sudo apt install ecss-restfs
```

13. Обновление пакета **ecss-media-server**.

Media Server анализирует конфигурацию в файлах находящихся в директории /etc/ecss/ecss-media-server/. В версии 3.17 реализована новая функция контроля приоритетности кодеков. Если по какой то причине в нескольких файлах присутствует секция "**codec**" в которой несколько раз прописывается **один и тот же кодек**, то после обновления возникнет конфликт и Media Server работать не будет.
Поэтому в конфигурационных файлах одного Media Server не должно быть конфликтов по приоритетам кодеков (запись об **одном** кодеке должна быть **одна** и не повторяться в других файлах). Какие именно фалы требуется сконфигурировать - на усмотрение администратора системы.

Предварительно отключить регистрацию на рабочий хост **ecss1**. Для этого надо закомментировать в конфигурационном файле строку с realm рабочего хоста (в примере realm="sip:10.0.20.81:5000"), следующей командой (добавив <!- - вначале строки , и - -> в конце строки), следующей командой:

на ecss2

```
sudo nano /etc/ecss/ecss-media-server/conf.d/default.xml
```

пример

```
<?xml version="1.0"?>
<config>
  <accounts>
          <!--<dynamic msr_name="msr.ecss2" realm="sip:10.0.20.81:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>-->
          <dynamic msr_name="msr.ecss2" realm="sip:10.0.20.82:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>
  </accounts>
</config>
```

Выполнить обновление:

на ecss2

```
sudo apt install ecss-media-server
```

Актуализировать конфигурацию:

на ecss2

```
sudo dpkg-reconfigure ecss-media-server
```

Выполнить проверку, и при необходимости отредактировать файл командой :

на ecss2

```
sudo nano /etc/ecss/ecss-media-server/config.xml
```

пример

пример

```
<?xml version="1.0" encoding="utf-8"?>
<config date="16:54:35 20.02.2025">
  <general log-level="3" log-rotate="yes" max-calls="8192" max-vid-calls="8192" max-in-group="512" load-sensor="media" load-delta="10" calls-delta="100" spool-dir-size="100M" log-name="msr.log" log-path="/var/log/ecss/media-server" use-srtp="disabled" enable-ice-transport="no" ice-update="no" aggressive-ice="yes" stun-server="" suspicious-mode="no"/>
  <transport bind-addr="10.0.20.80" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca" user="webrtc@live.com" password="muazkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60" rtcp-timeout="0" rtp-timeout="350" udp-src-check="no" cn-multiplier="3" port-start="12000" port-range="2048" tias-in-sdp="no" thread-cnt="2" vid-enc-threads="2" vid-dec-threads="2" video-conf-layout="evenly" keyframe-interval="1000" vid-decode-delay="100" silence-threshold="-30" dtmf-flash-disable="no" video-dscp="0" other-dscp="0" dummy-video-src="/usr/share/ecss-media-server/video/dummy_video.yuv" video-enc-width="640" video-enc-height="360" finalsilence="1000" rtcp-stat-dump="yes" dtmf-tg-fpc-loop="10" dtmf-tg-fit="1" dtmf-tg-fot="2" dtmf-tg-volume="12288"/>
  <codec pcma="1" pcmu="2" ilbc="0" gsm="0" g722="3" g726="0" g729="0" speex="0" l16="0" g7221="0" opus="0" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
  <accounts>
    <!-- <dynamic msr_name="msr.name"
            realm="sip:127.0.0.1:5000"
            dtmf_mode="rfc+inband+info"
            auth_name="user"
            auth_password="password" /> -->
  </accounts>
  <pbyte>
    <mcc bind-addr="10.0.20.82" port="5700"/>
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <metrics enable="no" msr_name="msr" port="8100"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
```

```
</config>
```

14. Обновление пакета **ecss-media-resources:**

на ecss2

```
sudo apt install ecss-media-resources
```

15. Обновление пакета **ecss-web-conf**

на ecss2

```
sudo apt install ecss-web-conf
```

16. **Обновление пакетов опциональных сервисов**

Обновление пакета **ecss-teleconference-ui**

на ecss2

```
sudo apt install ecss-teleconference-ui
```

Обновление пакета **ecss-crm-server**

на ecss2

```
sudo apt install ecss-crm-server
```

Обновление пакета **ecss-cc-ui**

на ecss2

```
sudo apt install ecss-cc-ui
```

Обновление пакета **ecss-call-api**

на ecss2

```
sudo apt install ecss-call-api
```

Обновление пакета **ecss-peeper-client**

на ecss2

```
sudo apt install ecss-peeper-client
```

17. Переносим актуальные регистрации абонентов:
Перед переводом keepalived на обновленный хост необходимо скопировать актуальную базу ds с ecss1 (иначе база регистраций будет пустая). Поэтому сначала делаем backup БД Mnesia на **ecss1**:

выполняется в CoCon

```
/node/ds1@ecss1/oasys/backup
```

```
/node/ds1@ecss1/oasys/backup
Backup was successfully created at the path: "/var/lib/ecss/oasys/
Mnesia.ds1@ecss1_20250930081517"
```

Останавливаем ecss-ds на ecss2:

на ecss2

```
sudo systemctl stop ecss-ds
```

Командой **scp** копируем backup с **ecss1** на **ecss2**.

на ecss2

```
scp -r ecss@ecss1:/var/lib/ecss/oasys/Mnesia.ds1@ecss1_20250930081517 /var/lib/ecss/oasys
```

Выполняем локальную копию текущей конфигурации:

на ecss2

```
sudo cp -a /var/lib/ecss/oasys/Mnesia.ds1@ecss2 /var/lib/ecss/oasys/Mnesia.ds1@ecss2-bkp
```

Заменяем локальную конфигурацию на конфигурацию backup с ecss1:

на ecss2

```
sudo mv /var/lib/ecss/oasys/Mnesia.ds1@ecss1_20250930081517 /var/lib/ecss/oasys/Mnesia.ds1@
ecss2
```

на ecss2

```
sudo chown -R ssw:ssw /var/lib/ecss/oasys
```

Запустить **ecss-ds** и **ecss-pa-sip** (чтобы подгрузились актуальные регистрации абонентов):

на ecss2

```
sudo systemctl start ecss-ds
```

на ecss2

```
sudo systemctl start ecss-pa-sip
```

Проверить статус сервисов командой **/system-status** :

выполняется в CoCon

```
/system-status
Checking...
┌─┬──────────────────┬──────────────────────────────┬──────────────────┬──────────────────┬──────┐
│ │       Node       │           Release            │   Erlang nodes   │ Mnesia nodes     │Uptime│
├─┼──────────────────┼──────────────────────────────┼──────────────────┼──────────────────┼──────┤
│ │core1@ecss2       │ecss-core-3.17.1.0.1771       │core1@ecss2       │not running       │2m 27s│
│ │ds1@ecss2         │ecss-ds-3.17.1.0.1771         │ds1@ecss2         │ds1@ecss2         │2m 32s│
│ │md1@ecss2         │ecss-mediator-3.17.1.0.1771   │md1@ecss2         │md1@ecss2         │2m 16s│
│ │mycelium1@ecss2   │ecss-mycelium-3.17.1.0.1771   │mycelium1@ecss2   │not running       │2m 39s│
│ │sip1@ecss2        │ecss-pa-sip-3.17.1.0.1771     │sip1@ecss2        │sip1@ecss2        │2m 22s│
│ │sorm1@ecss2       │ecss-sorm-3.17.1.0.1771       │sorm1@ecss2       │not running       │1m 56s│
└─┴──────────────────┴──────────────────────────────┴──────────────────┴──────────────────┴──────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────────┬─────────────┬──────────────┐
│     Node     │    MSR   │     MSR      │  Cc-status  │  Cc-uptime   │
│              │          │    version   │             │              │
├──────────────┼──────────┼──────────────┼─────────────┼──────────────┤
│ core1@ecss2  │ msr.ecss1 │ 3.14.16.1.0 │ connected   │ 00:00:10     │
│              │ msr.ecss2 │ 3.17.1.0.28 │ connected   │ 00:00:10     │
└──────────────┴──────────┴──────────────┴─────────────┴──────────────┘
```

Выполнить проверку переезда активных регистраций на обновленный хост. Для этого выполнить контроль зарегистрированы абонентов на одном(3.114.16) и втором (3.17) хостах. выполнить команду в CoCon:

выполняется в CoCon на ecss2

```
/domain/test_domain/sip/user/registered sip *
450 make users list ...
[**************************************************************************] 28mks
450 users information read ...
[**************************************************************************] 1ms
Executed on the sip1@ecss2
┌─────────────────┐
│ elements: 436 │
└─────────────────┘
```

выполняется в CoCon на ecss1

```
/domain/test_domain/sip/user/registered sip *
450 make users list ...
[**************************************************************************] 28mks
450 users information read ...
[**************************************************************************] 1ms
Executed on the sip1@ecss1
┌─────────────────┐
│ elements: 438 │
└─────────────────┘
```

18. Запуск сервиса **keepalived**

на ecss2

```
sudo systemctl enable --now keepalived
```

5 Подготовительные действия на ecss1

1. Подготовить **ecss1** к обновлению, для этого, перевести весь трафик на **ecss2** (команда выполняется на ecss1):

на ecss1

```
sudo systemctl disable --now keepalived.service
```

2. Командой **ifconfig** проверьте, что **все VRRP** адреса (**VRRP IP-mysql / VRRP IP-SIP1 / VRRP IP-SIP2**) находятся **на ecss2**.

на ecss2

```
ifconfig
```

3. После разделения кластера необходимо выключить все ноды на сервере **ecss1**:

на ecss1

```
sudo systemctl mask --now ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip ecss-sorm
```

4. Выполнить миграции таблиц DB подробное описание:
   Проверить, какие данные доступны для переноса, можно командой:

выполняется в CoCon

`/system/postgresql/migration/check <DATABASE> *`
где DATABASE:

- `address_book`
- `audit`
- `calls`
- `dialer`
- `history`
- `meetings`
- `subscribers`

Пример:

```
/system/postgresql/migration/check address_book * --username ******* --password *******
┌──────────────┬──────────────────┬──────────────────────────────────────────────┬──────────────────────┐
│      DB      │      Table       │                    Status                    │ Description          │
├──────────────┼──────────────────┼──────────────────────────────────────────────┼──────────────────────┤
│address_book  │contact           │Needs migration.                              │Depends on: `source`. │
│              │                  │Meegrated: 0 / 1785                           │                      │
│              │                  │                                              │                      │
│-             │label             │Needs migration.                              │No dependencies.      │
│              │                  │Meegrated: 0 / 5                              │                      │
│              │                  │                                              │                      │
│-             │phone             │Needs migration.                              │Depends on: `contact`, `label`.│
│              │                  │Meegrated: 0 / 2493                           │                      │
```

```
|            |               |                                    |
|
|-           |changes        |Needs migration.                    |No
dependencies.               |
|            |               |Meegrated: 0 / 5                    |
|
|            |               |                                    |
|
|-           |email          |Needs migration.                    |Depends on:
`contact`.         |
|            |               |Meegrated: 0 / 588                  |
|
└────────────┴───────────────┴────────────────────────────────────┴──────────
──────────────────┘
```

Миграцию выполнить командой:

выполняется в CoCon

```
/system/postgresql/migration/do <DATABASE> <TABLE>
```

Проверить прохождение миграции можно командой:

выполняется в CoCon

```
/system/postgresql/migration/info <DATABASE> <TABLE>
```

Пример:

```
/system/postgresql/migration/info address_book
┌────┬────────────┬──────────────────────┬──────────┬──────────────────┬──────────
──────┬──────────────────────┐
| ID |     DB     |       Table          |  Status  |     Props        |    Start
Time     |     End Time       |
├────┼────────────┼──────────────────────┼──────────┼──────────────────┼──────────
──────┼──────────────────────┤
|1   |address_book|source                |finished  |current: 2        |22.01.2026
13:14:20|22.01.2026 13:14:20|
|    |            |                      |          |overall: 2        |
|                  |
|2   |address_book|label                 |failed    |current: 0        |22.01.2026
13:14:30|22.01.2026 13:14:30|
|    |            |                      |          |overall: 5        |
|                  |
|3   |address_book|changes               |failed    |current: 0        |22.01.2026
13:14:49|22.01.2026 13:14:49|
|    |            |                      |          |overall: 5        |
|                  |
```

```
|4   |address_book|contact             |finished  |current: 1785   |22.01.2026
13:14:56|22.01.2026 13:14:56|
|    |            |                    |          |overall: 1785   |
|                  |
|5   |address_book|phone               |finished  |current: 2493   |22.01.2026
13:15:03|22.01.2026 13:15:03|
|    |            |                    |          |overall: 2493   |
|                  |
|6   |address_book|email               |finished  |current: 588    |22.01.2026
13:15:07|22.01.2026 13:15:07|
|    |            |                    |          |overall: 588    |
|                  |
└────┴────────────┴────────────────────┴──────────┴────────────────┴──────────────
──────┴────────────────────┘
```

6 Обновление сервисов SSW на ecss1

1. Выполнить на **ecss2** объединение в кластер:

   на ecss2

   ```
   sudo ecss-control attach
   ```

2. Активация сервиса **Keepalived**.

   на ecss1

   ```
   sudo systemctl enable --now keepalived
   ```

3. Обновление пакета **ecss-dns-env**.

   на ecss1

   ```
   sudo apt install ecss-dns-env
   ```

   При обновлении принимать новые конфигурационные файлы -Y.

   После обновления актуализировать конфигурацию командой:

   на ecss1

   ```
   sudo dpkg-reconfigure ecss-dns-env
   ```

4. Обновление пакета **ecss-user**:

на ecss1

```
sudo apt install ecss-user
```

5. Обновление пакета ecss-utils:

на ecss1

```
sudo apt install ecss-utils
```

6. Обновление пакета ecss-node:

на ecss1

```
sudo apt install ecss-node
```

на ecss1

ВОПРОСЫ НА УСТАНОВКУ ecss-node:
* ECSS-node ID in DNS format: <Согласно лицензии>.
* Default configuration: yes
* Global config: yes
!!!Принимаем все изменения файлов конфигурации.!!

7. Включить сервисы ssw:

на ecss1

```
sudo systemctl unmask ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip ecss-sorm
```

на ecss1

```
sudo systemctl enable ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip ecss-sorm
```

8. Сервис **ecss-sorm** на **ecss1** сразу замаскировать, т.к. у нас заявлена работа ecss-sorm **на одном хосте** в кластере.

на ecss1

```
sudo systemctl mask ecss-sorm
```

9. На **ecss1** запустите сервис **mycelium** и в **CoCon** проверьте, что сервис поднялся и видно ноды **mycelium** хостов ecss1/ ecss2:

на ecss1

```
sudo systemctl start ecss-mycelium
```

выполняется в CoCon

```
/system-status
Checking...
┌─┬──────────────┬──────────────────────────────────┬──────────────────────────────────┬──────────┐
│ │     Node     │          Release                 │         Erlang nodes             │Mnesia nodes│Uptime │
├─┼──────────────┼──────────────────────────────────┼──────────────────────────────────┼──────────┤
│ │core1@ecss2   │ecss-core-3.17.1.0.1771           │core1@ecss2                       │not running │3h 5m  │
│ │ds1@ecss2     │ecss-ds-3.17.1.0.1771             │ds1@ecss2                         │ds1@ecss2   │3h 5m  │
│ │md1@ecss2     │ecss-mediator-3.17.1.0.1771       │md1@ecss2                         │md1@ecss2   │3h 5m  │
│ │mycelium1@ecss1│ecss-mycelium-3.17.1.0.1771      │mycelium1@ecss1,mycelium1@ecss2   │not running │10m 42s│
│ │mycelium1@ecss2│ecss-mycelium-3.17.1.0.1771      │mycelium1@ecss1,mycelium1@ecss2   │not running │3h 5m  │
│ │sip1@ecss2    │ecss-pa-sip-3.17.1.0.1771         │sip1@ecss2                        │sip1@ecss2  │3h 5m  │
│ │sorm1@ecss2   │ecss-sorm-3.17.1.0.1771           │sorm1@ecss2                       │not running │3h 5m  │
└─┴──────────────┴──────────────────────────────────┴──────────────────────────────────┴──────────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────────┬────────────┬────────────┐
│     Node     │    MSR   │     MSR      │ Cc-status  │ Cc-uptime  │
│              │          │    version   │            │            │
├──────────────┼──────────┼──────────────┼────────────┼────────────┤
│ core1@ecss2  │ msr.ecss1 │ 3.14.16.1.0 │ connected  │ 00:21:09   │
│              │ msr.ecss2 │ 3.17.1.0.28 │ connected  │ 00:00:59   │
└──────────────┴──────────┴──────────────┴────────────┴────────────┘
```

10. Перед запуском **ecss-ds** на **ecss1** необходимо сделать backup директории /var/lib/ecss/oasys и удалить ее, для того чтобы с другого хоста загрузить актуальную БД. Сделать это можно переименовав существующую директорию:

После этого ecss-ds на хосте **ecss1** загрузит БД с хоста **ecss2**.

на ecss1

```
sudo mv /var/lib/ecss/oasys /var/lib/ecss/oasys-$(date +%Y%m%d-%H_%M)
```

11. На **ecss1** запустите сервис **ds** и в **CoCon** проверить, что сервис **ds** доступен на обоих хостах ecss1/ecss2:

на ecss1

```
sudo systemctl start ecss-ds
```

выполняется в CoCon

```
/system-status
Checking...
┌─┬──────────────┬──────────────────────────────┬──────────────────────────────────┬──────────────┬─────────┐
│ │     Node     │           Release            │          Erlang nodes            │ Mnesia nodes │ Uptime  │
├─┼──────────────┼──────────────────────────────┼──────────────────────────────────┼──────────────┼─────────┤
│ │core1@ecss2   │ecss-core-3.17.1.0.1771       │core1@ecss2                       │not running   │3h 13m   │
│ │ds1@ecss1     │ecss-ds-3.17.1.0.1771         │ds1@ecss1,ds1@ecss2               │ds1@ecss1,ds1@ecss2│12s   │
│ │ds1@ecss2     │ecss-ds-3.17.1.0.1771         │ds1@ecss1,ds1@ecss2               │ds1@ecss1,ds1@ecss2│3h 13m │
│ │md1@ecss2     │ecss-mediator-3.17.1.0.1771   │md1@ecss2                         │md1@ecss2     │3h 13m   │
│ │mycelium1@ecss1│ecss-mycelium-3.17.1.0.1771  │                                  │mycelium1@ecss1,mycelium1@ecss2│not running │18m 33s│
│ │mycelium1@ecss2│ecss-mycelium-3.17.1.0.1771  │                                  │mycelium1@ecss1,mycelium1@ecss2│not running │3h 13m │
│ │sip1@ecss2    │ecss-pa-sip-3.17.1.0.1771     │sip1@ecss2                        │sip1@ecss2    │3h 13m   │
│ │sorm1@ecss2   │ecss-sorm-3.17.1.0.1771       │sorm1@ecss2                       │not running   │3h 12m   │
└─┴──────────────┴──────────────────────────────┴──────────────────────────────────┴──────────────┴─────────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────────┬────────────┬────────────┐
│     Node     │    MSR   │  MSR version │ Cc-status  │ Cc-uptime  │
├──────────────┼──────────┼──────────────┼────────────┼────────────┤
│ core1@ecss2  │ msr.ecss1│ 3.14.16.1.0  │ connected  │ 00:29:01   │
```

```
|             | msr.ecss2 | 3.17.1.0.28   | connected | 00:08:50  |
└─────────────┴───────────┴───────────────┴───────────┴───────────┘
```

12. Далее такими же действиями запустить сервисы: **ecss-mediator**, **ecss-pa-sip**, **ecss-core** следующими командами:

на ecss1

```
sudo systemctl start ecss-core
```

на ecss1

```
sudo systemctl start ecss-mediator
```

на ecss1

```
sudo systemctl start ecss-pa-sip
```

Проверить статус в CoCom:

Результат:

```
/system-status
Checking...
┌─┬─────────────────┬──────────────────────────────┬──────────────────────────────────────┬──
──────────────────────┬──────┐
| |     Node        |          Release             |         Erlang nodes                 |
Mnesia nodes     |Uptime|
├─┼─────────────────┼──────────────────────────────┼──────────────────────────────────────┼──
──────────────────────┼──────┤
| |core1@ecss1     |ecss-core-3.17.1.0.1771       |core1@ecss1,core1@ecss2
|not running           |3m 50s|
| |core1@ecss2     |ecss-core-3.17.1.0.1771       |core1@ecss1,core1@ecss2
|not running           |3h 22m|
| |ds1@ecss1       |ecss-ds-3.17.1.0.1771         |ds1@ecss1,ds1@ecss2
|ds1@ecss1,ds1@ecss2   |9m 42s|
| |ds1@ecss2       |ecss-ds-3.17.1.0.1771         |ds1@ecss1,ds1@ecss2
|ds1@ecss1,ds1@ecss2   |3h 22m|
| |md1@ecss1       |ecss-mediator-3.17.1.0.1771   |md1@ecss1,md1@ecss2
|md1@ecss1,md1@ecss2   |5m 22s|
| |md1@ecss2       |ecss-mediator-3.17.1.0.1771   |md1@ecss1,md1@ecss2
|md1@ecss1,md1@ecss2   |3h 22m|
| |mycelium1@ecss1|ecss-mycelium-3.17.1.0.1771
|mycelium1@ecss1,mycelium1@ecss2|not running          |28m 4s|
| |mycelium1@ecss2|ecss-mycelium-3.17.1.0.1771
|mycelium1@ecss1,mycelium1@ecss2|not running          |3h 23m|
```

```
| |sip1@ecss1     |ecss-pa-sip-3.17.1.0.1771   |sip1@ecss1,sip1@ecss2
|sip1@ecss1,sip1@ecss2|4m 40s|
| |sip1@ecss2     |ecss-pa-sip-3.17.1.0.1771   |sip1@ecss1,sip1@ecss2
|sip1@ecss1,sip1@ecss2|3h 22m|
| |sorm1@ecss2    |ecss-sorm-3.17.1.0.1771     |sorm1@ecss2
|not running          |3h 22m|
└─┴────────────────┴──────────────────────────┴──────────────────────────────────┘
──────────────────────┴──────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────────┬──────────────┬──────────────┬──────────────┐
│     Node     │     MSR      │     MSR      │  Cc-status   │  Cc-uptime   │
│              │              │    version   │              │              │
├──────────────┼──────────────┼──────────────┼──────────────┼──────────────┤
│ core1@ecss1  │ msr.ecss1    │ 3.14.16.1.0  │ connected    │ 00:02:37     │
│ core1@ecss2  │ msr.ecss1    │ 3.14.16.1.0  │ connected    │ 00:38:31     │
│              │ msr.ecss2    │ 3.17.1.0.28  │ connected    │ 00:18:20     │
└──────────────┴──────────────┴──────────────┴──────────────┴──────────────┘
```

13. Обновление пакета **ecss-restfs**:

на ecss1

```
sudo apt install ecss-restfs
```

14. Обновление пакета **ecss-media-server:**

на ecss1

```
sudo apt install ecss-media-server
```

Актуализировать конфигурацию:

на ecss1

```
sudo dpkg-reconfigure ecss-media-server
```

Проверить и при необходимости исправить вручную файл → /etc/ecss/ecss-media-server/config.xml
пример

на ecss1

```
<?xml version="1.0" encoding="utf-8"?>
<config date="05:27:38 07.12.2024">
  <general log-level="3" log-rotate="yes" max-calls="8192" max-vid-
calls="8192" max-in-group="512" load-sensor="media" load-delta="10" calls-
delta="100" spool-dir-size="100M" log-name="msr.log" log-path="/var/log/ecss/
media-server" use-srtp="disabled" enable-ice-transport="no" ice-update="no"
aggressive-ice="yes" stun-server="" suspicious-mode="no"/>
  <transport bind-addr="10.0.20.82" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca" user="webrtc@live.com"
password="muazkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60" rtcp-
timeout="0" rtp-timeout="350" udp-src-check="no" cn-multiplier="3" port-
start="12000" port-range="2048" tias-in-sdp="no" thread-cnt="2" vid-enc-
threads="2" vid-dec-threads="2" video-conf-layout="evenly" keyframe-
interval="1000" vid-decode-delay="100" silence-threshold="-30" dtmf-flash-
disable="no" video-dscp="0" other-dscp="0" dummy-video-src="/usr/share/ecss-
media-server/video/dummy_video.yuv" video-enc-width="640" video-enc-
height="360" finalsilence="1000" rtcp-stat-dump="yes" dtmf-tg-fpc-loop="10"
dtmf-tg-fit="1" dtmf-tg-fot="2" dtmf-tg-volume="12288"/>
  <codec pcma="3" pcmu="2" ilbc="0" gsm="0" g722="3" g726="0" g729="1"
speex="5" l16="0" g7221="6" opus="0" h264="1" h263-1998="2" t38="1" tel-event-
pt="0"/>
  <accounts>
    <!-- <dynamic msr_name="msr.name"
            realm="sip:127.0.0.1:5000"
            dtmf_mode="rfc+inband+info"
            auth_name="user"
            auth_password="password" /> -->
  </accounts>
  <pbyte>
    <mcc bind-addr="10.0.20.82" port="5700"/>
  </pbyte>
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <metrics enable="no"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

Так же, для регистрации media-server`a с ecss2 на ecss1 **надо убрать** ранее добавленные комментирование из файла /etc/ecss/ecss-media-server/conf.d/default.xml на **ecss2**:

на ecss2

```
<?xml version="1.0"?>
<config>
  <accounts>
          <!--<dynamic msr_name="msr.ecss2" realm="sip:10.0.20.81:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>-->
          <dynamic msr_name="msr.ecss2" realm="sip:10.0.20.82:5000"
dtmf_mode="rfc+inband+info" auth_name="user" auth_password="password"/>
```

```
  </accounts>
</config>
```

После чего перезапустить сервис:

на ecss2

```
sudo systemctl restart ecss-media-server.service
```

15. Обновление пакета **ecss-media-resources:**

на ecss1

```
sudo apt install ecss-media-resources
```

16. Обновление пакета **ecss-web-conf**:

на ecss1

```
sudo apt install ecss-web-conf
```

17. **Обновление пакетов опциональных сервисов**

Обновление пакета **ecss-teleconference-ui**

на ecss1

```
sudo apt install ecss-teleconference-ui
```

Обновление пакета **ecss-crm-server**

на ecss1

```
sudo apt install ecss-crm-server
```

Обновление пакета **ecss-cc-ui**

на ecss1

```
sudo apt install ecss-cc-ui
```

Обновление пакета **ecss-call-api**

на ecss2

```
sudo apt install ecss-call-api
```

Обновление пакета **ecss-peeper-client**

на ecss2

```
sudo apt install ecss-peeper-client
```

7 Проверка обновления

1. После выполнения обновления проверить работу всех сервисов SSW.

16.6 Обновление с версии 3.17.1 на 3.18

- 1 Подготовка хостов к обновлению
- 2 Подготовка к обновлению сервисов SSW
- 3 Обновление сервисов SSW на ecss2
- 4 Подготовительные действия на ecss1
- 5 Обновление сервисов SSW на ecss1
- 7 Проверка обновления

16.6.1 1 Подготовка хостов к обновлению

1. В случае недоступности одной из нод кластера могут возникнуть проблемы с файлами. После восстановления работы они будут находиться в сплите, из-за чего собственноручно придется запускать синхронизацию между нодами.
   Для того чтобы решить данную проблему, потребуется воспользоваться ключом **cluster.favorite-child-policy**. При его включении все файлы, находящиеся в сплите, будут автоматически синхронизированы между собой по заданному правилу.
   Включение данного параметра производится командой:

   на ecss1

   ```
   sudo gluster volume set ecss_volume cluster.favorite-child-policy size
   ```

   Результат:

   ```
   sudo gluster volume set ecss_volume cluster.favorite-child-policy size
   volume set: success
   ```

Результат:

```
sudo gluster volume info

Volume Name: ecss_volume
Type: Replicate
Volume ID: 4bc875b1-f966-44e2-9b3f-04f3a7af83f3
Status: Started
Snapshot Count: 0
Number of Bricks: 1 x 2 = 2
Transport-type: tcp
Bricks:
Brick1: 10.0.10.51:/var/lib/ecss/glusterfs
Brick2: 10.0.10.52:/var/lib/ecss/glusterfs
Options Reconfigured:
cluster.favorite-child-policy: size
performance.client-io-threads: off
nfs.disable: on
transport.address-family: inet
storage.fips-mode-rchecksum: on
cluster.granular-entry-heal: on
```

2. При обновлении кластера следует обновлять сервера по очереди. Таким образом, при обновлении один из серверов будет всегда запущен и работа сервисов не будет прервана. Для разделения кластера следует использовать инструмент **ecss-control** на обоих серверах (Описание команд утилиты приведено в «Приложении Н»).
   Обмен sip сигнализацией происходит через SIP-VRRP IP адрес. Этот функционал реализован для возможности миграции Ip адреса на другой хост в случае падения/отключения/техобслуживания основного хоста. Контроль миграции IP адресов осуществляется сервисом **keepalive** на каждом хосте. Для перевода трафика на первый сервер (**ecss1**), необходимо выключить keepalive на втором хосте (**ecss2**). В этом случае оба SIP-VRRP IP адреса будут находится на первом хосте (**ecss1**). Выполнить следующую команду:

на ecss2

```
sudo systemctl disable --now keepalived.service
```

3. Командой **ifconfig** проверить, что **все VRRP IP-адреса** находятся на **ecss1 (VRRP-SIP1/VRRP-SIP2)**.

на ecss1

```
ifconfig
```

4. Необходимо замаскировать все ноды на сервере, который будет обновляться первым, отключаем автозапуск нод:

на ecss2

```
sudo systemctl mask --now ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip ecss-sorm
```

При выполнении приложенной команды будет уведомление.

```
Warning: The unit file, source configuration file or drop-ins of ecss-ds.service changed on disk. Run 'systemctl daemon-reload' to reload units.
Warning: The unit file, source configuration file or drop-ins of ecss-mycelium.service changed on disk. Run 'systemctl daemon-reload' to reload units.
Warning: The unit file, source configuration file or drop-ins of ecss-mediator.service changed on disk. Run 'systemctl daemon-reload' to reload units.
Warning: The unit file, source configuration file or drop-ins of ecss-core.service changed on disk. Run 'systemctl daemon-reload' to reload units.
Warning: The unit file, source configuration file or drop-ins of ecss-pa-sip.service changed on disk. Run 'systemctl daemon-reload' to reload units.
```

Данные сообщения не является ошибкой. Продолжать выполнение процедуры.

5. На **ecss1** выполнить разделение SSW на два независимых сервера следующей командой:

Команда **ecss-control** работает со стандартными именами хостов : **ecss1**/**ecss2**.

на ecss1

```
sudo ecss-control detach
```

Проверить статус:

на ecss1

```
sudo ecss-control status
```

Результат:

```
sudo ecss-control detach
detach host: ecss2, ip: 10.0.10.52
complete
```

```
sudo ecss-control status
locked ip: 10.0.10.52
```

Нода **sorm** идет одна на кластер. В случае её работы на **ecss1**, при выполнении приложенной команды будет уведомление.

```
Unit /etc/systemd/system/ecss-sorm.service is masked, ignoring.
```

Данное сообщение не является ошибкой. Продолжать выполнение процедуры.

16.6.2 2 Подготовка к обновлению сервисов SSW

1. Актуализировать репозиторий ssw:

на обоих хостах

```
sudo sh -c "echo deb [arch=amd64]  http://archive.eltex.org/ssw/jammy/3.18 stable main extras external > /etc/apt/sources.list.d/eltex-ecss10-stable.list"
```

2. Выполнить обновление пакетов:

на обоих хостах

```
sudo apt update
```

16.6.3 3 Обновление сервисов SSW на ecss2

1. Обновление PostgreSQL
   a. Проверить на каких адресах подключена БД Postgres, выполнить команду:

на ecss2

```
psql -Upostgres -decss_storekeeper_db -h localhost -p5439 -c "SELECT node_local_dsn FROM bdr.bdr_nodes;"
```

Пример: пароль по умолчанию "postgres1"

```
psql -Upostgres -decss_storekeeper_db -h localhost -p5439 -c "SELECT node_local_dsn FROM bdr.bdr_nodes;"
Password for user postgres:
```

```
                                        node_local_dsn
-------------------------------------------------------------------------------
---
 host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db user=postgres
password=postgres1
 host=10.0.10.52 port=5439 dbname=ecss_storekeeper_db user=postgres
password=postgres1
(2 rows)
```

b. Выключаем docker контейнер

на ecss2

```
sudo docker compose -f /srv/ecss/ecss-postgres-bdr-ssw/docker-compose.container.yml
down
```

Пример:

```
sudo docker compose -f /srv/ecss/ecss-postgres-bdr-ssw/docker-
compose.container.yml down
[+] down 2/2
 ✓ Container ecss-postgres-bdr-ssw-postgres-container-1 Removed


 ✓ Network ecss-postgres-bdr-ssw_default                Removed
```

c. Удаляем старый deb пакет, т.к опция upgrade не поддержана

на ecss2

```
sudo apt remove ecss-postgres-bdr-ssw
```

!!Важно!!

Выполнить именно **remove** а не purge, т.к remove сохраняет директорию volumes с таблицами нашей БД

Пример:

```
sudo apt remove ecss-postgres-bdr-ssw
Чтение списков пакетов… Готово
. . .
Следующие пакеты будут УДАЛЕНЫ:
  ecss-postgres-bdr-ssw
. . .
```

```
Хотите продолжить? [Д/н] y
Удаляется ecss-postgres-bdr-ssw (17.1.0.59+ssw) …
```

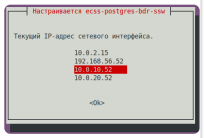
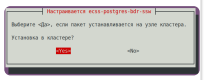
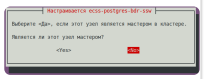
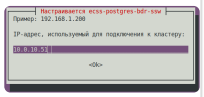
d. Устанавливаем новый deb пакет

на ecss2

```
sudo apt install ecss-postgres-bdr-ssw
```

!!Важно!!

Во время установки, будут заданы вопросы как при первоначальной установке:
Ответы должны **точно соответствовать** указанным в процедуре, иначе будет выполнена попытка создать новую базу данных и установка завершится ошибкой

Вопрос	Ответ	Пример
Текущий IP-адрес сетевого интерфейса. Выберите IP: Такой-же как и при первоначальной установке, т.к репликация завязана на конкретные IP (проверили в пункте "a", в примере 10.0.10.52)	10.0.10.52 (Необходимо выбрать)	
Установка в кластере? Выберите <Да>, если пакет устанавливается на узле кластера.	Да (значение по умолчанию)	
Является ли этот узел мастером? НЕТ! (!!Важно!! иначе он попытается создать новую bdr, чего сделать не сможет и установка завершиться с ошибкой)	Нет (значение по умолчанию)	
IP-адрес, используемый для подключения к кластеру: IP соседней ноды (который указывали при её установке проверяли в пункте "a", в примере 10.0.10.51)	10.0.10.51 (необходимо указать)	

При успешной установке, на этапе поднятия репликации увидим:

```
sudo apt install ecss-postgres-bdr-ssw
. . .
============================
 Checking replication status
```

```
============================
Репликация BDR настроена корректно: данные совпадают.
. . .
=========================================================
=========================================================
 ecss-postgres-bdr-ssw successfully installed/updated!
=========================================================
=========================================================


⌐ ⌐ ⌐ ¬ ⌐ ¬ ⌐ ⌐ ¬ ¬ ¬ ⌐ ⌐ ¬ ⌐
| | | ╰ | ┌ ┼ ╰ ┼ | | ┼ ╰ ╰ | |
┘ ┘ ┘ ┴ ┘ ┘ ┘ ┘ ┴ ┴ ┴ ┘ ┘ ┘ ┘
```

e. После установки, проверить, что подтянулся новый image и стартовал docker, выполнить команду

на ecss2

```
docker ps
```

Пример:

```
docker ps
CONTAINER ID   IMAGE                                                COMMAND
CREATED         STATUS         PORTS                                                NAMES
80ef72d3b125   git.eltex.loc:4567/ecss/ecss-postgresbdr:3.18   "/docker-
entrypoint.…"   8 minutes ago   Up 8 minutes   0.0.0.0:5439->5432/tcp, [::]:5439->54
32/tcp   ecss-postgres-bdr-ssw-postgres-container-1
```

2. Обновление SSW.
   a. Обновление пакета ecss-dns-env:

на ecss2

```
sudo apt install ecss-dns-env
```

После обновления, нужно актуализировать конфигурацию, выполнив команду. Ответы на вопросы аналогичны процедуре инсталляции:

на ecss2

```
sudo dpkg-reconfigure ecss-dns-env
```

   b. Выполнить обновление пакета ecss-user:

на ecss2

```
sudo apt install ecss-user
```

c. Выполнить обновление пакета ecss-utils:

на ecss2

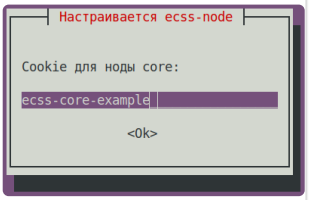
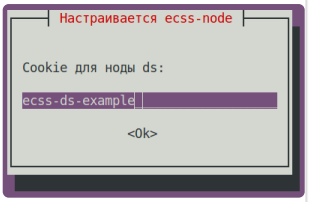
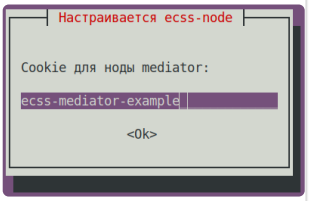
```
sudo apt install ecss-utils
```

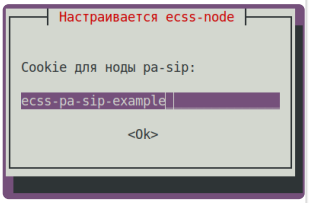
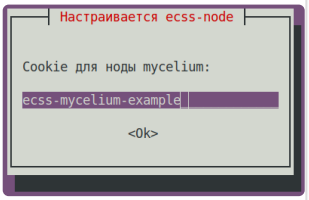
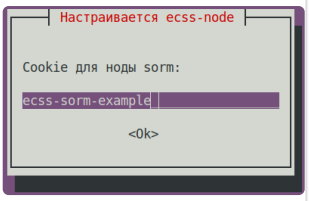
d. Выполнить обновление пакета ecss-node:

на ecss2

```
sudo apt install ecss-node
```

ВОПРОСЫ НА УСТАНОВКУ ecss-node:

Вопрос	Ответ	Пример
Cookie для ноды core: (указать уникальный cookie для core , в примере ecss-core-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-core-example** (необходимо ввести)	
Cookie для ноды ds: (указать уникальный cookie для ds , в примере ecss-ds-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-ds-example** (необходимо ввести)	
Cookie для ноды mediator: (указать уникальный cookie для mediator , в примере ecss-mediator-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mediator-example** (необходимо ввести)	

Вопрос	Ответ	Пример
Cookie для ноды pa-sip: (указать уникальный cookie для pa-sip , в примере ecss-pa-sip-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-pa-sip-example** (необходимо ввести)	
Cookie для ноды mycelium: (указать уникальный cookie для mycelium , в примере ecss-mycelium-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mycelium-example** (необходимо ввести)	
Cookie для ноды sorm: (указать уникальный cookie для sorm , в примере ecss-sorm-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-sorm-example** (необходимо ввести)	

!!!Принимаем все изменения файлов конфигурации.!!

3. Включаем ноды.
   Включить сервисы ssw следующими командами:

на ecss2

```
sudo systemctl unmask ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip ecss-sorm
```

на ecss2

```
sudo systemctl enable ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip ecss-sorm
```

Внимание

Перед запуском нод SSW, изменяем имя кластера (на любое другое, главное чтобы оно не совпадало с соседом (ecss1)/любой другой потенциальной АТС в вашей сети, иначе он попробует к ней подключится). Для этого правим в файле /etc/ecss/ecss-mycelium/mycelium.config строчку " {cluster_name, '**eltex.test**'}," для примера на " {cluster_name, '**eltex1.test**'},"

на ecss2

```
sudo nano /etc/ecss/ecss-mycelium/mycelium1.config
```

Пример:

```
[
    {mycelium_broker, [
        {cluster_name, 'eltex1.test'},
        {compress_module, elx_data_nif},
        {session_gc_timeout, 3000},
        {queue_gc_timeout, infinity}
. . .
```

4. Запуск SSW на **ecss2**. Запустить следующие сервисы:
ecss-mycelium

на ecss2

```
sudo systemctl start ecss-mycelium
```

ecss-ds

на ecss2

```
sudo systemctl start ecss-ds
```

ecss-core

на ecss2

```
sudo systemctl start ecss-core
```

ecss-mediator

на ecss2

```
sudo systemctl start ecss-mediator
```

- Если лицензия поддерживает СОРМ3, то запустить также сервис ecss-sorm.

  на ecss2

  ```
  sudo systemctl start ecss-sorm
  ```

  Если сервис **ecss-sorm** оказался **замаскирован**, то выполнить следующие команды:

  на ecss2

  ```
  sudo systemctl unmask ecss-sorm
  sudo systemctl start ecss-sorm
  ```

Для корректной работы SORM3 необходимо выполнить коррекцию команды для sorm экстрактора. **Данную команду необходимо выполнить для каждого домена в SSW.**

выполняется в CoCon

```
/domain/<Имя_домена>/sorm3/set extractor_node sorm1@ecss2
```

Для примера:

```
/domain/test_domain/sorm3/info
┌──────────────────┬─────────────┬──────────────┐
│     Property     │  Domain     │   Value      │
├──────────────────┼─────────────┼──────────────┤
│adaptation        │test_domain  │undefined     │
│enabled           │test_domain  │false         │
│entrypoint_profile│test_domain  │              │
│extractor_ip      │test_domain  │127.0.0.1     │
│extractor_node    │test_domain  │sorm1@ecss1   │
└──────────────────┴─────────────┴──────────────┘
/domain/test_domain/sorm3/set extractor_node sorm1@ecss2
Property "extractor_node" successfully changed from:
sorm1@ecss1
   to
sorm1@ecss2.
/domain/test_domain/sorm3/info

┌──────────────────┬─────────────┬──────────────┐
│     Property     │  Domain     │   Value      │
├──────────────────┼─────────────┼──────────────┤
│adaptation        │test_domain  │undefined     │
│enabled           │test_domain  │false         │
│entrypoint_profile│test_domain  │              │
│extractor_ip      │test_domain  │127.0.0.1     │
│extractor_node    │test_domain  │sorm1@ecss2   │
└──────────────────┴─────────────┴──────────────┘
```

Команду **повторить для всех доменов** в SSW

!!!ПОКА НЕ ЗАПУСКАЕМ PA-SIP!!!

5. Проверить статус сервисов, для этого выполнить в CoCon команду system-status:

выполняется в CoCon

```
/system-status
Checking...
```

	Node	Release	Erlang nodes	Mnesia nodes	Uptime
	core1@ecss2	ecss-core-3.18.0.217	core1@ecss2	not running	1m 28s
	ds1@ecss2	ecss-ds-3.18.0.217	ds1@ecss2	ds1@ecss2	1m 32s
	md1@ecss2	ecss-mediator-3.18.0.217	md1@ecss2	md1@ecss2	1m 23s
	mycelium1@ecss2	ecss-mycelium-3.18.0.217	mycelium1@ecss2	not running	1m 37s
	sorm1@ecss2	ecss-sorm-3.18.0.217	sorm1@ecss2	not running	1m 56s

```
All services are started.

  Active media resource selected list specific:
```

Node	MSR	MSR version	Cc-status	Cc-uptime
core1@ecss2	msr.ecss1	3.17.1.0.27	connected	00:00:14
	msr.ecss2	3.17.1.0.27	connected	00:00:16

6. Обновление пакета **ecss-restfs**.

Принять **все новые** конфигурационные файлы, выбирая **Y**.

на ecss2

```
sudo apt install ecss-restfs
```

7. Обновление пакета **ecss-media-server**.

> ⚠ Если использовались не дефолтные имена файлов конфигурации, то при установке создадутся (при отсутствии) новые файлы с дефолтными именами и при запуске media-server будет использовать именно их.
> В случае использования дефолтных имен, в процессе установки **ecss-media-server** выполнит backup конфигурационных файлов, создав файлы с расширением **.dpkg-old**. Во вновь созданных файлах конфигурации, часть конфигураций окажется удаленной/измененной на дефолтные значения (например ice параметры необходимые для elph, stun сервер, адрес сервера с метриками).
> Поэтому при необходимости скорректировать параметры перед вводом media-server в эксплуатацию, добавив из backup конфиг файлов *.dpkg-old нужные данные.
> Если использовались не дефолтные имена файлов конфигурации, новосозданные **дефолтные файлы нужно в обязательно порядке удалить**.

Выполнить обновление:

на ecss2

```
sudo apt install ecss-media-server
```

✅ Принимаем все новые файлы конфигурации

Актуализировать конфигурацию:

на ecss2

```
sudo dpkg-reconfigure ecss-media-server
```

8. Обновление пакета **ecss-media-resources:**

на ecss2

```
sudo apt install ecss-media-resources
```

9. Обновление пакета **ecss-web-conf**

на ecss2

```
sudo apt install ecss-web-conf
```

10. **Обновление пакетов опциональных сервисов**

Проверить список установленных сервисов командой:

на ecss2

```
sudo dpkg –l | grep ecss
```

Пример:

```
sudo dpkg -l | grep ecss
ii  ecss-cc-ui                              17.1.0.36+cc-ui
amd64        Eltex/ECSS-10 UI and API for call center
ii  ecss-dns-env                            18.0.0
amd64        Eltex/ECSS-10 Configuration settings for domain service
ii  ecss-media-resources                     18.0.0
amd64        Eltex/ECSS-10 Media resources
ii  ecss-media-server                        18.0.7
amd64        Eltex/ECSS-10 Media server
ii  ecss-node                               18.0.217
amd64        Eltex/ECSS-10 Node services
ii  ecss-peeper-client                       17.1.0.23
amd64        Eltex/ECSS-10 peeper client
ii  ecss-postgres-bdr-ssw                    18.0.0+ssw
amd64        Eltex/Postgres BDR
ii  ecss-restfs                              18.0.21
amd64        Eltex/ECSS-10 RestFS API interface for filesystem
ii  ecss-teleconference-ui                   17.1.0.23
amd64        Eltex/ECSS-10 UI and API for Teleconference
ii  ecss-user                               18.0.0
amd64        Eltex/ECSS-10 SSW user environment
ii  ecss-utils                              18.0.0
amd64        Eltex/ECSS-10 Utilities for reading binary logs
ii  ecss-web-conf                           18.0.31
amd64        Eltex/ECSS-10 Web configurator
```

По результатам выполнения команды переустановить пакеты имеющие версию 17.1.* . В данном примере – ecss-cc-ui/ecss-peeper-client/ecss-teleconference-ui.

Обновление пакета **ecss-teleconference-ui**

на ecss2

```
sudo apt install ecss-teleconference-ui
```

Обновление пакета **ecss-crm-server**

на ecss2

```
sudo apt install ecss-crm-server
```

Обновление пакета **ecss-cc-ui**

на ecss2

```
sudo apt install ecss-cc-ui
```

Обновление пакета **ecss-call-api**

на ecss2

```
sudo apt install ecss-call-api
```

Обновление пакета **ecss-peeper-client**

на ecss2

```
sudo apt install ecss-peeper-client
```

11. Запуск сервиса **keepalived**
    Перед запуском сервиса keepalived, необходимо стартовать сервис ecss-pa-sip:

на ecss2

```
sudo systemctl start ecss-pa-sip
```

на ecss2

```
sudo systemctl enable --now keepalived
```

16.6.4 4 Подготовительные действия на ecss1

1. Подготовить **ecss1** к обновлению, для этого, перевести весь трафик на **ecss2** (команда выполняется на ecss1):

на ecss1

```
sudo systemctl disable --now keepalived.service
```

2. Командой **ifconfig** проверьте, что **все VRRP** адреса (**VRRP IP-SIP1 / VRRP IP-SIP2**) находятся **на ecss2**.

на ecss2

```
ifconfig
```

3. После разделения кластера необходимо выключить все ноды на сервере **ecss1**:

на ecss1

```
sudo systemctl mask --now ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip ecss-sorm
```

При выполнении приложенной команды будет уведомление.

```
Warning: The unit file, source configuration file or drop-ins of ecss-ds.service changed on disk. Run 'systemctl daemon-reload' to reload units.
Warning: The unit file, source configuration file or drop-ins of ecss-mycelium.service changed on disk. Run 'systemctl daemon-reload' to reload units.
Warning: The unit file, source configuration file or drop-ins of ecss-mediator.service changed on disk. Run 'systemctl daemon-reload' to reload units.
Warning: The unit file, source configuration file or drop-ins of ecss-core.service changed on disk. Run 'systemctl daemon-reload' to reload units.
Warning: The unit file, source configuration file or drop-ins of ecss-pa-sip.service changed on disk. Run 'systemctl daemon-reload' to reload units.
```

Данные сообщения не является ошибкой. Продолжать выполнение процедуры.

16.6.5 5 Обновление сервисов SSW на ecss1

1. Выполнить на **ecss1** объединение в кластер:

Команда **ecss-control attach** выполняется на этом шаге для возможности создания репликации БД Postgres. Такое поведение безопасно, потому что нода уже выведена из эксплуатации.

на ecss1

```
sudo ecss-control attach
```

2. Активация сервиса **keepalived**.

VRRP-SIP завязан на сервиса ecss-pa-sip. Если демон ecss-pa-sip выключен, адрес не мигрирует на ecss1.

на ecss1

```
sudo systemctl enable --now keepalived
```

3. Обновление PostgreSQL
    a. Проверить на каких адресах подключена БД Postgres, выполнить команду:

на ecss1

```
psql -Upostgres -decss_storekeeper_db -h localhost -p5439 -c "SELECT node_local_dsn
FROM bdr.bdr_nodes;"
```

Пример: пароль по умолчанию "postgres1"

```
psql -Upostgres -decss_storekeeper_db -h localhost -p5439 -c "SELECT node_local_dsn
FROM bdr.bdr_nodes;"
Password for user postgres:
                                    node_local_dsn
-------------------------------------------------------------------------------
---
 host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db user=postgres
password=postgres1
 host=10.0.10.52 port=5439 dbname=ecss_storekeeper_db user=postgres
password=postgres1
(2 rows)
```

    b. Выключаем docker контейнер

на ecss1

```
sudo docker compose -f /srv/ecss/ecss-postgres-bdr-ssw/docker-compose.container.yml
down
```

Пример:

```
sudo docker compose -f /srv/ecss/ecss-postgres-bdr-ssw/docker-
compose.container.yml down
[+] down 2/2
 ✓ Container ecss-postgres-bdr-ssw-postgres-container-1 Removed


 ✓ Network ecss-postgres-bdr-ssw_default                Removed
```

    c. Удаляем старый deb пакет, т.к опция upgrade не поддержана

на ecss1

```
sudo apt remove ecss-postgres-bdr-ssw
```

!!Важно!!

Выполнить именно **remove** а не purge, т.к remove сохраняет директорию volumes с таблицами нашей БД

Пример:

```
sudo apt remove ecss-postgres-bdr-ssw
Чтение списков пакетов… Готово
. . .
Следующие пакеты будут УДАЛЕНЫ:
  ecss-postgres-bdr-ssw
. . .
Хотите продолжить? [Д/н] y
Удаляется ecss-postgres-bdr-ssw (17.1.0.59+ssw) …
```

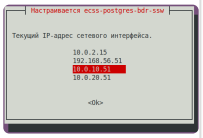
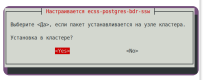
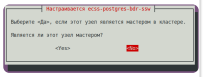
d. Устанавливаем новый deb пакет

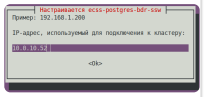
на ecss1

```
sudo apt install ecss-postgres-bdr-ssw
```

!!Важно!!

Во время установки, будут заданы вопросы как при первоначальной установке:
Ответы должны **точно соответствовать** указанным в процедуре, иначе будет выполнена попытка создать новую базу данных и установка завершится ошибкой

Вопрос	Ответ	Пример
Текущий IP-адрес сетевого интерфейса. Выберите IP: Такой-же как и при первоначальной установке, т.к репликация завязана на конкретные IP (проверили в пункте "a", в примере 10.0.10.51)	10.0.10.51 (Необходимо выбрать)	
Установка в кластере? Выберите <Да>, если пакет устанавливается на узле кластера.	Да (значение по умолчанию)	
Является ли этот узел мастером? НЕТ! (!!Важно!! иначе он попытается создать новую bdr, чего сделать не сможет и установка завершиться с ошибкой)	Нет (значение по умолчанию)	

Вопрос	Ответ	Пример
IP-адрес, используемый для подключения к кластеру: IP соседней ноды (который указывали при её установке проверяли в пункте "a", в примере 10.0.10.52)	10.0.10.52 (необходимо указать)	

При успешной установке, на этапе поднятия репликации увидим:

```
sudo apt install ecss-postgres-bdr-ssw
. . .
============================
 Checking replication status
============================
Репликация BDR настроена корректно: данные совпадают.
. . .
============================================================
============================================================
 ecss-postgres-bdr-ssw successfully installed/updated!
============================================================
============================================================


```

e. После установки, проверить, что подтянулся новый image и стартовал docker, выполнить команду

на ecss1

```
docker ps
```

Пример:

```
docker ps
CONTAINER ID   IMAGE                                                COMMAND
CREATED         STATUS         PORTS                                                NAMES
f53bd28c78ff   git.eltex.loc:4567/ecss/ecss-postgresbdr:3.18   "/docker-
entrypoint.…"   3 minutes ago   Up 3 minutes   0.0.0.0:5439->5432/tcp, [::]:5439->54
32/tcp   ecss-postgres-bdr-ssw-postgres-container-1
```

4. Обновление пакета **ecss-dns-env**.

на ecss1

```
sudo apt install ecss-dns-env
```

При обновлении принимать новые конфигурационные файлы -Y.

После обновления актуализировать конфигурацию командой:

на ecss1

```
sudo dpkg-reconfigure ecss-dns-env
```

5. Обновление пакета **ecss-user**:

на ecss1

```
sudo apt install ecss-user
```

6. Обновление пакета ecss-utils:

на ecss1

```
sudo apt install ecss-utils
```

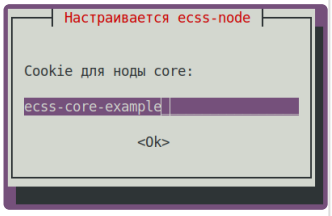
7. Обновление пакета ecss-node:

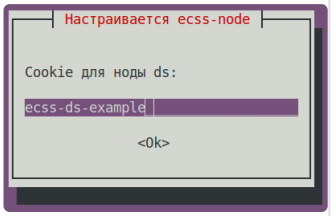
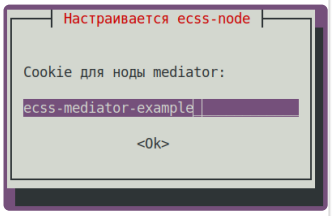
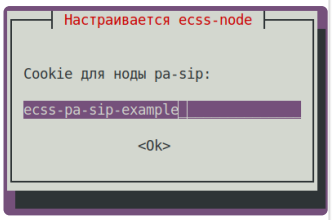
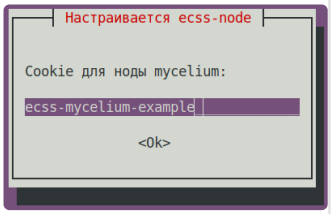
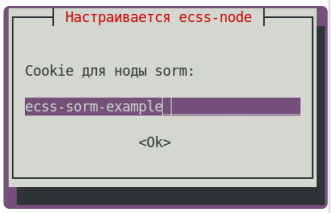
на ecss1

```
sudo apt install ecss-node
```

на ecss1

ВОПРОСЫ НА УСТАНОВКУ ecss-node:

Вопрос	Ответ	Пример
Cookie для ноды core: (указать уникальный cookie для core , в примере ecss-core-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-core-example** (необходимо ввести)	

Вопрос	Ответ	Пример
Cookie для ноды ds: (указать уникальный cookie для ds , в примере ecss-ds-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-ds-example** (необходимо ввести)	
Cookie для ноды mediator: (указать уникальный cookie для mediator , в примере ecss-mediator-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mediator-example** (необходимо ввести)	
Cookie для ноды pa-sip: (указать уникальный cookie для pa-sip , в примере ecss-pa-sip-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-pa-sip-example** (необходимо ввести)	
Cookie для ноды mycelium: (указать уникальный cookie для mycelium , в примере ecss-mycelium-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-mycelium-example** (необходимо ввести)	
Cookie для ноды sorm: (указать уникальный cookie для sorm , в примере ecss-sorm-example) В кластерной конфигурации cooke должны быть одинаковые для нод одного типа	**ecss-sorm-example** (необходимо ввести)	

!!!Принимаем все изменения файлов конфигурации.!!

8. Включить сервисы ssw:

на ecss1

```
sudo systemctl unmask ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip ecss-sorm
```

на ecss1

```
sudo systemctl enable ecss-ds ecss-mycelium ecss-mediator ecss-core ecss-pa-sip ecss-sorm
```

Внимание

Перед запуском нод SSW, изменяем имя кластера (на любое другое, главное чтобы оно не совпадало с соседом (ecss1)/любой другой потенциальной АТС в вашей сети, иначе он попробует к ней подключится). Для этого правим в файле /etc/ecss/ecss-mycelium/mycelium.config строчку " {cluster_name, '**eltex.test**'}," для примера на " {cluster_name, '**eltex1.test**'},"

на ecss1

```
sudo nano /etc/ecss/ecss-mycelium/mycelium1.config
```

Пример:

```
[
    {mycelium_broker, [
        {cluster_name, 'eltex1.test'},
        {compress_module, elx_data_nif},
        {session_gc_timeout, 3000},
        {queue_gc_timeout, infinity}
. . .
```

9. Сервис **ecss-sorm** на **ecss1** сразу замаскировать, т.к. у нас заявлена работа ecss-sorm **на одном хосте** в кластере.

на ecss1

```
sudo systemctl mask ecss-sorm
```

10. На **ecss1** запустите сервис **mycelium** и в **CoCon** проверьте, что сервис поднялся и видно ноды **mycelium** хостов ecss1/ ecss2:

на ecss1

```
sudo systemctl start ecss-mycelium
```

выполняется в CoCon

```
/system-status
Checking...
┌─┬─────────────────┬──────────────────────────────┬──────────────────────────────────────┬──────────────────┬────────┐
│ │      Node       │           Release            │             Erlang nodes             │  Mnesia nodes    │Uptime │
├─┼─────────────────┼──────────────────────────────┼──────────────────────────────────────┼──────────────────┼────────┤
│ │core1@ecss2      │ecss-core-3.18.0.249          │core1@ecss1,core1@ecss2               │not running       │12m 12s│
│ │ds1@ecss2        │ecss-ds-3.18.0.249            │ds1@ecss1,ds1@ecss2                   │ds1@ecss1,ds1@ecss2  │12m 12s│
│ │md1@ecss2        │ecss-mediator-3.18.0.249      │md1@ecss1,md1@ecss2                   │md1@ecss1,md1@ecss2  │12m 12s│
│ │mycelium1@ecss1  │ecss-mycelium-3.18.0.249      │mycelium1@ecss1,mycelium1@ecss2       │not running       │12m 14s│
│ │mycelium1@ecss2  │ecss-mycelium-3.18.0.249      │mycelium1@ecss1,mycelium1@ecss2       │not running       │12m 13s│
│ │sip1@ecss2       │ecss-pa-sip-3.18.0.249        │sip1@ecss1,sip1@ecss2                 │sip1@ecss1,sip1@ecss2│12m 13s│
└─┴─────────────────┴──────────────────────────────┴──────────────────────────────────────┴──────────────────┴────────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────────┬────────────┬─────────────┐
│     Node     │   MSR    │     MSR      │ Cc-status  │ Cc-uptime   │
│              │          │   version    │            │             │
├──────────────┼──────────┼──────────────┼────────────┼─────────────┤
│ core1@ecss2  │ msr.ecss1 │ 3.17.1.0.27 │ connected  │ 00:10:50    │
│              │ msr.ecss2 │ 3.18.0.7    │ connected  │ 00:10:51    │
└──────────────┴──────────┴──────────────┴────────────┴─────────────┘
```

11. На **ecss1** запустите сервис **ds** и в **CoCon** проверить, что сервис **ds** доступен на обоих хостах ecss1/ecss2:

на ecss1

```
sudo systemctl start ecss-ds
```

выполняется в CoCon

```
/system-status
Checking...
┌─┬─────────────────┬──────────────────────────────┬──────────────────────────────────────┬──────────────────┬────────┐
│ │      Node       │           Release            │             Erlang nodes             │  Mnesia nodes    │Uptime │
├─┼─────────────────┼──────────────────────────────┼──────────────────────────────────────┼──────────────────┼────────┤
```

```
──────────────────┼────────┤
| |core1@ecss2    |ecss-core-3.18.0.249    |core1@ecss1,core1@ecss2        |not
running          |12m 12s|
| |ds1@ecss1      |ecss-ds-3.18.0.249      |ds1@ecss1,ds1@ecss2
|ds1@ecss1,ds1@ecss2  |12m 14s|
| |ds1@ecss2      |ecss-ds-3.18.0.249      |ds1@ecss1,ds1@ecss2
|ds1@ecss1,ds1@ecss2  |12m 12s|
| |md1@ecss2      |ecss-mediator-3.18.0.249|md1@ecss1,md1@ecss2
|md1@ecss1,md1@ecss2  |12m 12s|
| |mycelium1@ecss1|ecss-mycelium-3.18.0.249|mycelium1@ecss1,mycelium1@ecss2|not
running          |12m 14s|
| |mycelium1@ecss2|ecss-mycelium-3.18.0.249|mycelium1@ecss1,mycelium1@ecss2|not
running          |12m 13s|
| |sip1@ecss2     |ecss-pa-sip-3.18.0.249  |sip1@ecss1,sip1@ecss2
|sip1@ecss1,sip1@ecss2|12m 13s|
└─┴─────────────────┴────────────────────────┴──────────────────────────────────┴─────
──────────────────┴────────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────────┬────────────┬──────────────┐
│     Node     │    MSR   │     MSR      │ Cc-status  │ Cc-uptime    │
│              │          │   version    │            │              │
├──────────────┼──────────┼──────────────┼────────────┼──────────────┤
│ core1@ecss2  │ msr.ecss1 │ 3.17.1.0.27 │ connected  │ 00:10:50     │
│              │ msr.ecss2 │ 3.18.0.7    │ connected  │ 00:10:51     │
└──────────────┴──────────┴──────────────┴────────────┴──────────────┘
```

12. Далее такими же действиями запустить сервисы: **ecss-mediator**, **ecss-pa-sip**, **ecss-core** следующими командами:

на ecss1

```
sudo systemctl start ecss-core
```

на ecss1

```
sudo systemctl start ecss-mediator
```

на ecss1

```
sudo systemctl start ecss-pa-sip
```

Проверить статус в CoCom:

Результат:

```
/system-status
Checking...
┌─┬────────────────┬──────────────────────────────┬──────────────────────────────────────┬──────────────────┬────────┐
│ │     Node       │         Release              │          Erlang nodes                │ Mnesia nodes     │Uptime  │
├─┼────────────────┼──────────────────────────────┼──────────────────────────────────────┼──────────────────┼────────┤
│ │core1@ecss1     │ecss-core-3.18.0.249          │core1@ecss1,core1@ecss2               │not running       │12m 14s │
│ │core1@ecss2     │ecss-core-3.18.0.249          │core1@ecss1,core1@ecss2               │not running       │12m 12s │
│ │ds1@ecss1       │ecss-ds-3.18.0.249            │ds1@ecss1,ds1@ecss2                   │ds1@ecss1,ds1@ecss2 │12m 14s │
│ │ds1@ecss2       │ecss-ds-3.18.0.249            │ds1@ecss1,ds1@ecss2                   │ds1@ecss1,ds1@ecss2 │12m 12s │
│ │md1@ecss1       │ecss-mediator-3.18.0.249      │md1@ecss1,md1@ecss2                   │md1@ecss1,md1@ecss2 │12m 14s │
│ │md1@ecss2       │ecss-mediator-3.18.0.249      │md1@ecss1,md1@ecss2                   │md1@ecss1,md1@ecss2 │12m 12s │
│ │mycelium1@ecss1 │ecss-mycelium-3.18.0.249      │mycelium1@ecss1,mycelium1@ecss2       │not running       │12m 14s │
│ │mycelium1@ecss2 │ecss-mycelium-3.18.0.249      │mycelium1@ecss1,mycelium1@ecss2       │not running       │12m 13s │
│ │sip1@ecss1      │ecss-pa-sip-3.18.0.249        │sip1@ecss1,sip1@ecss2                 │sip1@ecss1,sip1@ecss2 │12m 14s │
│ │sip1@ecss2      │ecss-pa-sip-3.18.0.249        │sip1@ecss1,sip1@ecss2                 │sip1@ecss1,sip1@ecss2 │12m 13s │
└─┴────────────────┴──────────────────────────────┴──────────────────────────────────────┴──────────────────┴────────┘

All services are started.

  Active media resource selected list specific:
┌──────────────┬──────────┬──────────────┬────────────┬─────────────┐
│     Node     │    MSR   │     MSR      │ Cc-status  │ Cc-uptime   │
│              │          │   version    │            │             │
├──────────────┼──────────┼──────────────┼────────────┼─────────────┤
│ core1@ecss1  │ msr.ecss1 │ 3.17.1.0.27 │ connected  │ 00:10:51    │
│              │ msr.ecss2 │ 3.18.0.7    │ connected  │ 00:10:48    │
│ core1@ecss2  │ msr.ecss1 │ 3.17.1.0.27 │ connected  │ 00:10:50    │
│              │ msr.ecss2 │ 3.18.0.7    │ connected  │ 00:10:51    │
└──────────────┴──────────┴──────────────┴────────────┴─────────────┘
```

13. Обновление пакета **ecss-restfs**:

на ecss1

```
sudo apt install ecss-restfs
```

14. Обновление пакета **ecss-media-server:**

на ecss1

```
sudo apt install ecss-media-server
```

Актуализировать конфигурацию:

на ecss1

```
sudo dpkg-reconfigure ecss-media-server
```

Проверить и при необходимости исправить вручную файл → /etc/ecss/ecss-media-server/config.xml
пример

на ecss1

```
<?xml version="1.0" encoding="utf-8"?>
<config date="05:27:38 07.12.2024">
  <general log-level="3" log-rotate="yes" max-calls="8192" max-vid-calls="8192" max-in-group="512" load-sensor="media" load-delta="10" calls-delta="100" spool-dir-size="100M" log-name="msr.log" log-path="/var/log/ecss/media-server" use-srtp="disabled" enable-ice-transport="no" ice-update="no" aggressive-ice="yes" stun-server="" suspicious-mode="no"/>
  <transport bind-addr="10.0.20.82" port="5040" transport="udp+tcp"/>
  <!-- By default configured public TURN-server -->
  <turn-server use-turn="no" host="numb.viagenie.ca" user="webrtc@live.com" password="muazkh"/>
  <media mixer-clock-rate="8000" use-vad="no" cng-level="0" jb-size="60" rtcp-timeout="0" rtp-timeout="350" udp-src-check="no" cn-multiplier="3" port-start="12000" port-range="2048" tias-in-sdp="no" thread-cnt="2" vid-enc-threads="2" vid-dec-threads="2" video-conf-layout="evenly" keyframe-interval="1000" vid-decode-delay="100" silence-threshold="-30" dtmf-flash-disable="no" video-dscp="0" other-dscp="0" dummy-video-src="/usr/share/ecss-media-server/video/dummy_video.yuv" video-enc-width="640" video-enc-height="360" finalsilence="1000" rtcp-stat-dump="yes" dtmf-tg-fpc-loop="10" dtmf-tg-fit="1" dtmf-tg-fot="2" dtmf-tg-volume="12288"/>
  <codec pcma="3" pcmu="2" ilbc="0" gsm="0" g722="3" g726="0" g729="1" speex="5" l16="0" g7221="6" opus="0" h264="1" h263-1998="2" t38="1" tel-event-pt="0"/>
  <accounts>
    <!-- <dynamic msr_name="msr.name"
            realm="sip:127.0.0.1:5000"
            dtmf_mode="rfc+inband+info"
            auth_name="user"
            auth_password="password" /> -->
  </accounts>
  <pbyte>
    <mcc bind-addr="10.0.20.82" port="5700"/>
  </pbyte>
```

```
  <conf_dir path="/etc/ecss/ecss-media-server/conf.d"/>
  <metrics enable="no"/>
  <rtp>
    <auto addr-v4=""/>
  </rtp>
</config>
```

15. Обновление пакета **ecss-media-resources:**

на ecss1

```
sudo apt install ecss-media-resources
```

16. Обновление пакета **ecss-web-conf**:

на ecss1

```
sudo apt install ecss-web-conf
```

17. **Обновление пакетов опциональных сервисов**

Проверить список установленных сервисов командой:

на ecss2

```
sudo dpkg -l | grep ecss
```

Пример:

```
sudo dpkg -l | grep ecss
ii  ecss-cc-ui                              17.1.0.53+cc-ui
amd64        Eltex/ECSS-10 UI and API for call center
ii  ecss-dns-env                            18.0.0
amd64        Eltex/ECSS-10 Configuration settings for domain service
ii  ecss-license-master                     1.0.77
amd64        Eltex/ECSS-10 license master
ii  ecss-media-resources                    18.0.0
amd64        Eltex/ECSS-10 Media resources
ii  ecss-media-server                       18.0.7
amd64        Eltex/ECSS-10 Media server
ii  ecss-node                               18.0.249
amd64        Eltex/ECSS-10 Node services
ii  ecss-peeper-client                      17.1.0.23
amd64        Eltex/ECSS-10 peeper client
ii  ecss-postgres-bdr-ssw                   18.0.0+ssw
amd64        Eltex/Postgres BDR
```

```
ii  ecss-restfs                          18.0.22
amd64        Eltex/ECSS-10 RestFS API interface for filesystem
ii  ecss-teleconference-ui               17.1.0.31
amd64        Eltex/ECSS-10 UI and API for Teleconference
ii  ecss-user                            18.0.0
amd64        Eltex/ECSS-10 SSW user environment
ii  ecss-utils                           18.0.0
amd64        Eltex/ECSS-10 Utilities for reading binary logs
ii  ecss-web-conf                        18.0.31
amd64        Eltex/ECSS-10 Web configurator
```

По результатам выполнения команды переустановить пакеты имеющие версию 17.1.* . В данном примере – ecss-cc-ui/ecss-peeper-client/ecss-teleconference-ui.

Обновление пакета **ecss-teleconference-ui**

на ecss1

```
sudo apt install ecss-teleconference-ui
```

Обновление пакета **ecss-cc-ui**

на ecss1

```
sudo apt install ecss-cc-ui
```

Обновление пакета **ecss-peeper-client**

на ecss2

```
sudo apt install ecss-peeper-client
```

16.6.6 7 Проверка обновления

1. После выполнения обновления проверить работу всех сервисов SSW.

17 Начальная настройка конфигурации

- Понятия и определения
- Предварительные условия
- Настройка ECSS-10 для производительных систем
  - Выделение отдельных ядер процессора для MSR
  - Установка работы CPU в режим perfomance
    - Выставить параметры ОС в режим производительности
  - Запуск MSR на изолированных ядрах процессора
  - Настройка использования определенных ядер процессора для erlang-based служб
- Порядок начальной настройки конфигурации
  - Настройка программного медиасервера
  - Создание и настройка группы SIP-IP-адресов (IP-set)
    - Пример создание через CLI (в примере имя IP set "test_set", IP add = 10.0.10.91/92)
    - Пример создание через Web (в примере имя IP set "test_set", IP add = 10.0.10.91/92) приложение "Кластеры"
  - Создание и настройка доменов (виртуальных АТС)
    - Декларация доменов
      - Пример создание через CLI (в примере имя домена "test_test"):
      - Пример создание через Web (в примере имя домена "test_test") приложение "Домены":
        - Создание домена
        - Подключение IP set
    - Настройка маршрутизации
      - Создание контекстов маршрутизации
      - Применение контекстов маршрутизации для системных интерфейсов
    - Создание и настройка абонентов
    - Создание и настройка транков
    - Создание и настройка bridge-интерфейсов
    - Настройка ограничений
    - Сценарии IVR
  - Настройка дополнительных сервисов
    - Настройка взаимодействия с серверами RADIUS
    - Настройка функции СОРМ
    - Настройка дополнительных приложений

В данном разделе приведено описание этапов первоначальной настройки ECSS перед проверкой базового функционала.

17.1 Понятия и определения

- **Алиас** — совокупность данных об абоненте;
- **Бридж** — виртуальный шлюз, объединяющий связь между виртуальными АТС. Понятие «бридж» было введено для создания средств контроля соединениями между виртуальными АТС. Вызовы между виртуальными АТС одной системы ECSS-10 маршрутизируются в рамках данной системы через бридж. При этом не задействованы межстанционные соединительные линии. Бридж представлен в виде двух связанных друг с другом интерфейсов. Каждый интерфейс декларирован в своей виртуальной АТС. Для бриджа, как и для классического транка, могут быть заданы различные типы ограничений, например, количество каналов, что дает ограничение на количество одновременно установленных соединений между виртуальными АТС и позволяет нормировать нагрузку;
- **Домен (виртуальная АТС)** — совокупность, состоящая из множества контекстов маршрутизации, интерфейсов и алиасов. Ближайший эквивалент — описание плана нумерации и маршрутизации в рамках классической телефонной станции для традиционных сетей;

- **Кластер** — совокупность элементов одного типа, выполняющих, с точки зрения системы, единую функцию. С их помощью описывается вычислительная топология системы. В нашей системе элементом кластера является **нода.** Кластер существует до тех пор, пока в его состав входит хотя бы одна нода;
- **Медиаресурсы** — описание параметров медиасервера, необходимых для работы с ним;
- **Медиасервер (MSR)** — компонент системы ECSS-10, предназначенный для проксирования речевой и видеоинформации по протоколу RTP, организации конференций, записи разговоров, воспроизведения медиафайлов и различных комбинаций этих режимов. Управление ресурсами медиасервера осуществляется с помощью механизма *control channel* (RFC 6230 Media Control Channel Framework, RFC 6231 IVR Control Package, RFC 6505 Mixer Control Package);
- **Нода** — представляет собой виртуальную машину Erlang и является элементом вычислительного кластера ECSS-10. Ноды в ECSS-10 типизируются по выполняемому на них функционалу. Однотипные ноды объединяются в кластеры соответствующего типа. Пример: кластер Core состоит из нод, выполняющих функцию ядра коммутационной системы;
- **СОРМ** — система технических средств для обеспечения функций оперативно-розыскных мероприятий;
- **IVR** (англ. **Interactive Voice Response**), интерактивное голосовое меню — система предварительно записанных голосовых сообщений, выполняющая функцию маршрутизации звонков внутри call-центра или УПАТС с использованием информации, вводимой клиентом на клавиатуре телефона с помощью тонального набора;
- **LDAP** (англ. **Lightweight Directory Access Protocol** — «легковесный протокол доступа к каталогам») — протокол прикладного уровня для доступа к службе каталогов;
- **RADIUS** — протокол, который предоставляет централизованный метод аутентификации пользователей путем обращения к внешнему серверу. Протокол RADIUS используется для аутентификации, авторизации и учета. Сервер RADIUS использует базу данных пользователей, которая содержит данные проверки подлинности для каждого пользователя. Таким образом, использование протокола RADIUS обеспечивает централизованное управление и дополнительную защиту при доступе к ресурсам сети.

17.2 Предварительные условия

Перед началом настройки конфигурации необходимо убедиться в следующем:

- установлена операционная система;
- настроен доступ в сеть Интернет;
- установлено необходимое ПО;
- установлены необходимые компоненты ECSS;
- настроен сервис NTP;
- подключен и работает Token;
- все подсистемы ECSS запущены и работают;
- установлены корректные паспорт и лицензия;
- если система в кластере, то дополнительно:
    - настроен и корректно работает keepalived;
    - работает репликация БД PostgreSQL;
    - ноды в работе и "видят" друг друга;
    - настроен и работает GlusterFS.

17.3 Настройка ECSS-10 для производительных систем

Для производительных систем настройка ECSS-10 состоит из следующих этапов:

17.3.1 Выделение отдельных ядер процессора для MSR

Для того чтобы изолировать MSR-медиасервер от остальной системы, необходимо выделить под него отдельные ядра процессора. Для выделения отдельных ядер процессора необходимо выполнить следующие действия:

1. Отредактировать файл grub:

   ```
   sudo nano /etc/default/grub
   ```

   Привести параметр GRUB_CMDLINE_LINUX="" к следующему виду:

   Данный пример изолирует ядра с 8 по 11. Также возможен вариант с перечислением 1, 2, 4-6 и т.п.

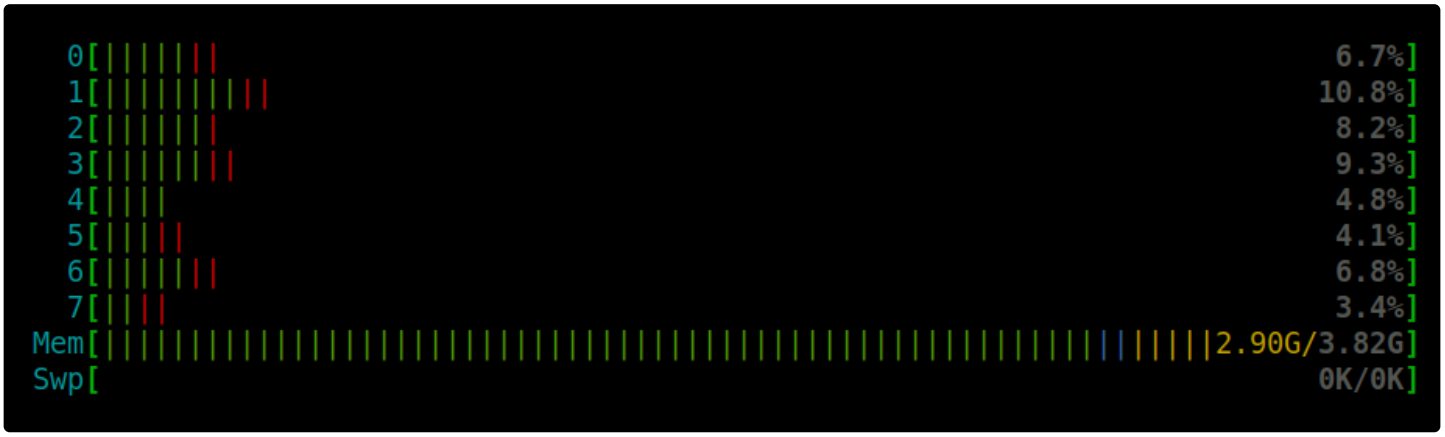
   ```
   GRUB_CMDLINE_LINUX="isolcpus=0-4"
   ```

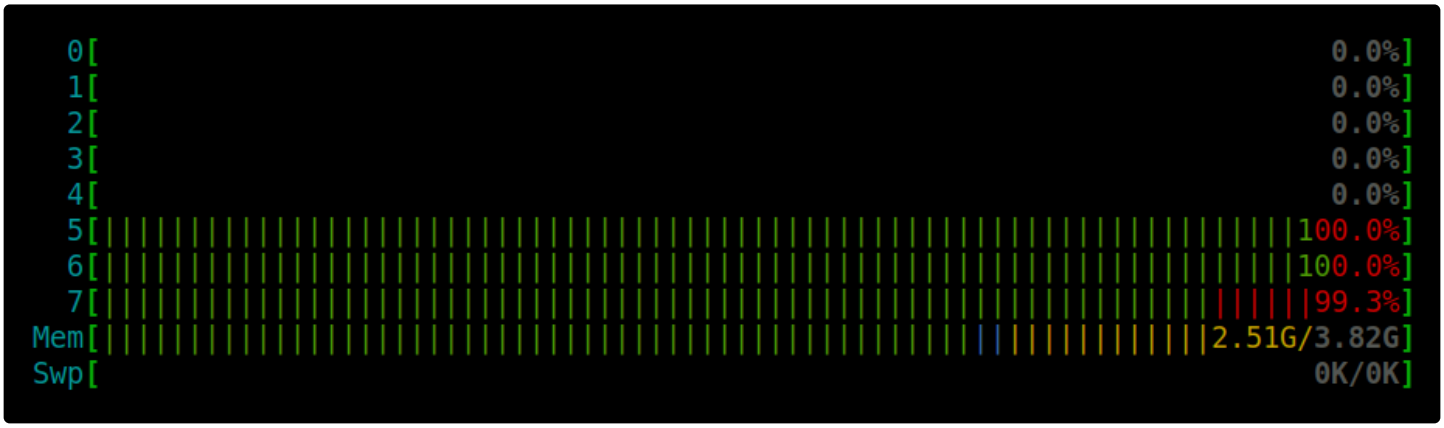
2. Обновить конфигурацию grub. Для этого выполните команду:

   ```
   sudo update-grub
   ```

3. Перезапустить систему.
   Если всё сделано правильно, то после перезагрузки на изолированных ядрах htop будет показывать нулевую нагрузку.

Пример (Было / Стало):





17.3.2 Установка работы CPU в режим perfomance

По умолчанию в Ubuntu есть пять профилей работы процессора.

Описание профилей:

- **conservative** — медленно повышает частоту процессора в зависимости от нагрузки на систему и резко сбрасывает частоту к минимальной при простое;
- **ondemand** — быстро повышает частоту процессора при возрастании нагрузки и медленно сбрасывает частоту к минимуму при простое;
- **userspace** — позволяет указывать частоту вручную;
- **powersave** — соответствует минимальной допустимой частоте CPU;
- **performance** — соответствуют максимальной частоте CPU.

Выставить параметры ОС в режим производительности

Используем утилиту cpufrequtils.

```
sudo apt install cpufrequtils
```

По умолчанию после инсталляции Ubuntu использует режим "ondemand" - "по запросу" (производительность CPU по запросу приложений, экономит электроэнергию , но ниже производительность):

```
cat /etc/init.d/cpufrequtils | grep GOVERNOR=
```

Результат: режим работы по умолчанию после инсталляции - "ondemand"

```
#     GOVERNOR="ondemand"
GOVERNOR="ondemand"
```

Установить режим - результативность/производительность - в файле /etc/init.d/cpufrequtils значение "**ondemand**" заменить на "**performance**"

```
sudo nano /etc/init.d/cpufrequtils
```

Перезапустить утилиту:

```
sudo /etc/init.d/cpufrequtils restart
```

Затем выполнить команду:

```
sudo systemctl daemon-reload
```

17.3.3 Запуск MSR на изолированных ядрах процессора

Для того чтобы MSR запускался на отдельных ядрах процессора, необходимо:

1. Включить экземпляр MSR:

   ```
   systemctl enable ecss-media-server@msr.service
   systemctl edit ecss-media-server@msr.service
   ```

2. Привести файл /etc/systemd/system/ecss-media-server.service.d/override.conf к следующему виду:

   ```
   [Service]
   CPUAffinity=8-11
   CPUSchedulingPolicy=rr
   ```

Для просмотра того, на каких ядрах запустился сервис, можно воспользоваться htop. В нем нужно добавить колонку Processor.

В данном примере MSR запущен на ядрах 8, 9, 10, 11. CPUSchedulingPolicy необходим, только если указан isolcpus.

Настройка MSR более подробно описана на странице Настройка медиасервера (MSR).

Аналогичным образом настраивается конфигурация ядер для CORE/SIP/SIGTRAN/DS/MD

17.3.4 Настройка использования определенных ядер процессора для erlang-based служб

Для того чтобы ядра процессора использовались правильно, необходимо скорректировать параметры запуска erlang-нод на производительных системах.

Для этого разрабатывается схема размещения нод на ядрах.

Схема разрабатывается по следующим правилам:

- использовать более двух ядер;
- необходимо, чтобы одна нода не использовала ядра на разных процессорах;
- для сильно нагруженных нод, таких как core и sip, нужно выделять индивидуальные ядра;
- ноды, которые не загружены, можно размещать на одном ядре;
- для core необходимо выделять большее количество ядер.

Распределения ядер на примере двухпроцессорного сервера HP BL660 с двумя процессорами Intel Xeon E5-4657L с 12 ядрами и поддержкой гипертрединга, которые могут образовать 8 виртуальных ядер:

```
myc  0-3
ds   4-7
core 8-23
sip  24-31
md   32-35
rest 36-39
sp   40-43
msr  44-47
```

Для осуществления данного распределения необходимо включить режим использования только необходимого количества ядер на erlang-ноде.

Для этого редактируем файл vm.args каждой ноды, расположенный по пути **/usr/lib/ecss/ECSS-SERVICE-NAME/releases/VERSION/**.

Например, для ecss-core необходимо отредактировать файл:

```
sudo nano /usr/lib/ecss/ecss-core/releases/3.14.2.29/vm.args
```

В этот файл добавить опции, которые задают использование требуемого количества логических ядер процессора и количество активных шедулеров.

Для 16-ти ядер это:

```
+sct L0-15c0-15
+sbt db
+S16:16
```

Для 8-ми ядер:

```
+sct L0-7c0-7
+sbt db
+S8:8
```

Для 4-х ядер:

```
+sct L0-3c0-3
+sbt db
+S4:4
```

Для 2-х ядер:

```
+sct L0-1c0-1
+sbt db
+S2:2
```

Следующая задача — установить сервис на выбранные ядра. Делается это аналогично тому, как описано для MSR.

Необходимо выполнить команду:

```
sudo systemctl edit ECSS-SERVICE-NAME
```

Далее добавить параметры:

```
[Service]
CPUAffinity=0-3
```

где в значении CPUAffinity указываются те ядра, на которых должны запускаться процессы сервиса.

Пример настройки ecss-core по указанной выше схеме:

```
> sudo systemctl edit ecss-core.service
[Service]
CPUAffinity=8-23
```

После настройки параметров CPUAffinity для всех сервисов необходимо перезагрузить конфигурацию услуг командой:

```
sudo systemctl daemon-reload
```

Перезапустить сервисы:

```
sudo systemctl restart ecss.slice
```

Убедиться в корректной привязке сервисов к ядрам можно утилитой htop, включив отображение колонки PROCESSOR.

17.4 Порядок начальной настройки конфигурации

В первую очередь необходимо выполнить общие настройки для всей системы:

- медиасервер;
- IP-set
- Создание и настройка доменов
- маршрутизация, модификация и адаптация;
- пользователи ECSS;
- опционально:
    - COPM;
    - RADIUS;
    - LDAP.

Далее необходимо задекларировать и настроить следующие сервисы:

- домены;
- транки;
- бриджи;
- IVR;
- абоненты;
- услуги на транках и абонентах.

В последнюю очередь настраиваются дополнительные сервисы.

Рекомендуется проводить первоначальную настройку в порядке, приведенном ниже.

17.4.1 Настройка программного медиасервера

Порядок настройки программного медиасервера (MSR):

- настройка конфигурационного файла медиасервера;
- запуск медиасервера;
- добавление медиаресурсов на ECSS-10:
  - командами CLI (см. раздел "/system/media/resource/ - команды управления медиаресурсами");
  - через приложение web-конфигуратора "Сетевые окончания MSR (MSR registrars)".
- опционально в соответствии с проектом:
  - настройка распределенного медиасервера;
  - замена видеозаставки.

17.4.2 Создание и настройка группы SIP-IP-адресов (IP-set)

Произвести настройку адаптера SIP можно выполнить одним из двух способов:

- через интерфейс командной строки (CLI);
- через приложение web-конфигуратора "Кластеры".

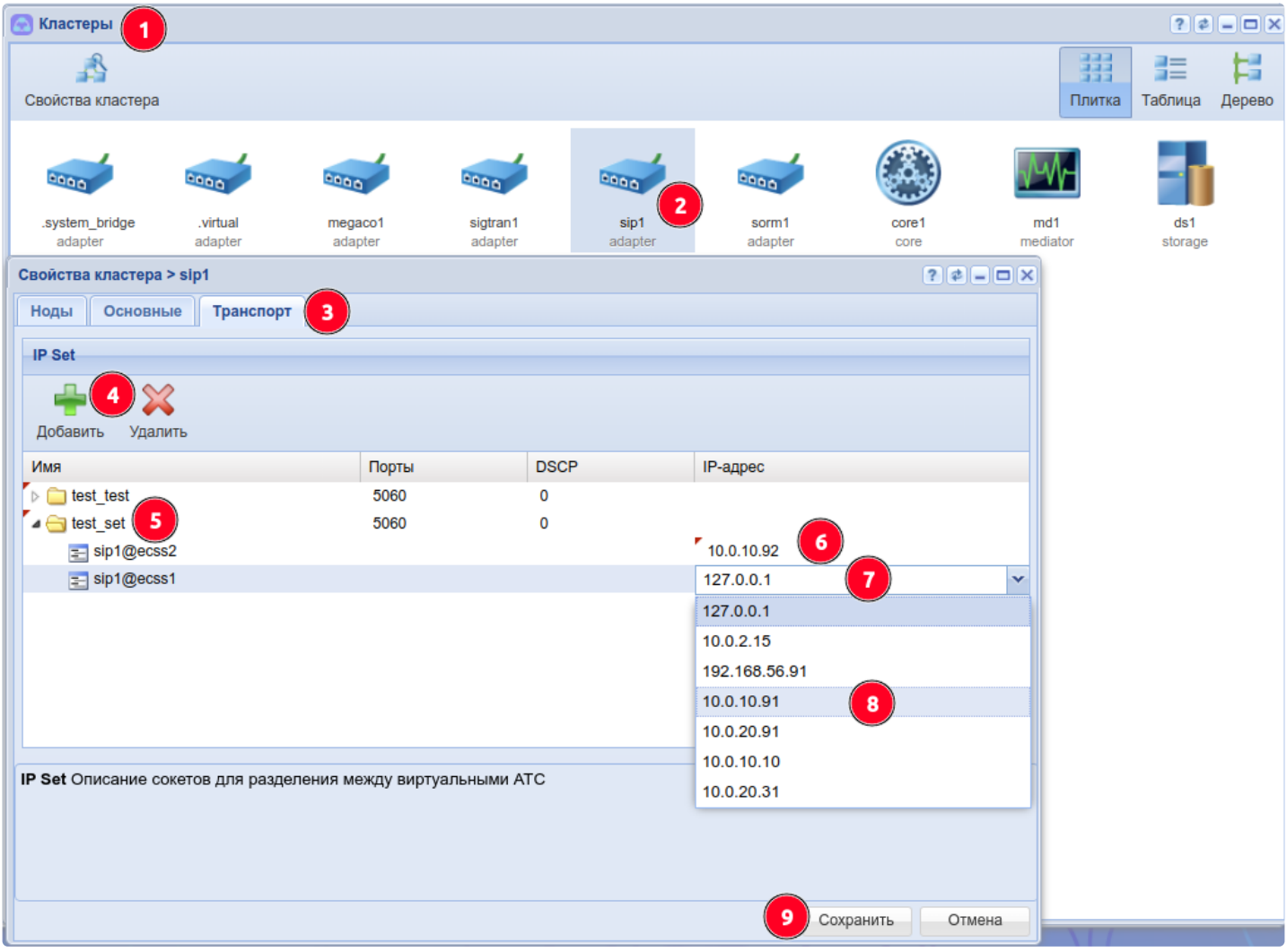
Без предварительного создания IP-set не будут работать команды создания Домена/транка/абонента. Для работы данных объектов обязательным условием является информация, через какие IP адреса (порты) будет выполняться обмен сигнальной информацией. IP-set может быть как индивидуальный для каждого домена, так и общим для всех доменов SSW, в зависимости от проекта. Так же можно назначить на разные домены индивидуальные порты (для примера домен А использует порт=5061, а домен В порт=5062) на одних IP адресах (10.0.10.91/92)

Пример создание через CLI (в примере имя IP set "test_set", IP add = 10.0.10.91/92)

1. Создать IP set и указать IP адрес для первой ноды:
   `/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss1 ip = 10.0.10.91`
2. К созданному IP set добавить IP адрес второй ноды:
   `/cluster/adapter/sip1/sip/network/set ip_set test_set node-ip node = sip1@ecss2 ip = 10.0.10.92`
3. К созданному IP set добавить IP порт:
   `/cluster/adapter/sip1/sip/network/set ip_set test_set listen-ports list = [5060]`

Пример создание через Web (в примере имя IP set "test_set", IP add = 10.0.10.91/92) приложение "Кластеры"

1. Открыть приложение "Кластеры";
2. Двойным кликом открыть "sip1 adapter";
3. В окне Свойства кластера "sip1" выбрать "Транспорт";
4. Добавить;
5. Указать имя нового IP set (в примере "test_set");
6. Из выпадающего списка выбрать IP адрес, через который будет работать ecss2 нод;
7. Так же указать IP адрес для ecss1 нод (по умолчанию подставляет 127.0.0.1);
8. Выбрать правильный (в примере 10.0.10.91);
9. Сохранить.



17.4.3 Создание и настройка доменов (виртуальных АТС)

Данный этап включает в себя процесс создания виртуальных АТС, настройку правил маршрутизации вызовов, транков, абонентов, правил обслуживания абонентов.

Вся инфраструктура предоставления услуг телефонной связи на базе ECSS-10, а именно конфигурация подключаемых шлюзов, абонентские данные, план нумерации и правила маршрутизации, а также права доступа к функциям операционного управления и поддержки описываются в рамках определенного домена.

Таким образом, домен можно представить как логическую часть гибкого коммутатора, реализующую функционал отдельной АТС.
Таких сущностей на гибком коммутаторе может быть несколько. В системе ECSS-10 домен и виртуальная АТС — синонимы.
Фактически развертывание нескольких доменов и связей между ними дает возможность реализации сегмента или всей сети NGN в рамках одной инсталляции.
Системы доменов и гибкая система разграничения прав доступа позволяет оператору связи выполнять функции хостинга АТС для сторонних заказчиков.
Заказчик оператора связи может разместить свою корпоративную УПАТС или узел связи на мощностях системы ECSS-10, развернутой у оператора. При этом функции операционного управления за данной АТС могут быть переданы заказчику полностью или частично (используется схема разграничения ответственности за эксплуатацией данной АТС).

Декларация доменов

Необходимо создать домен по каждому из запроектированных доменов (виртуальных АТС).

Создание домена можно выполнить одним из двух способов:

- через интерфейс командной строки (CLI);
- через приложение web-конфигуратора "Домены".

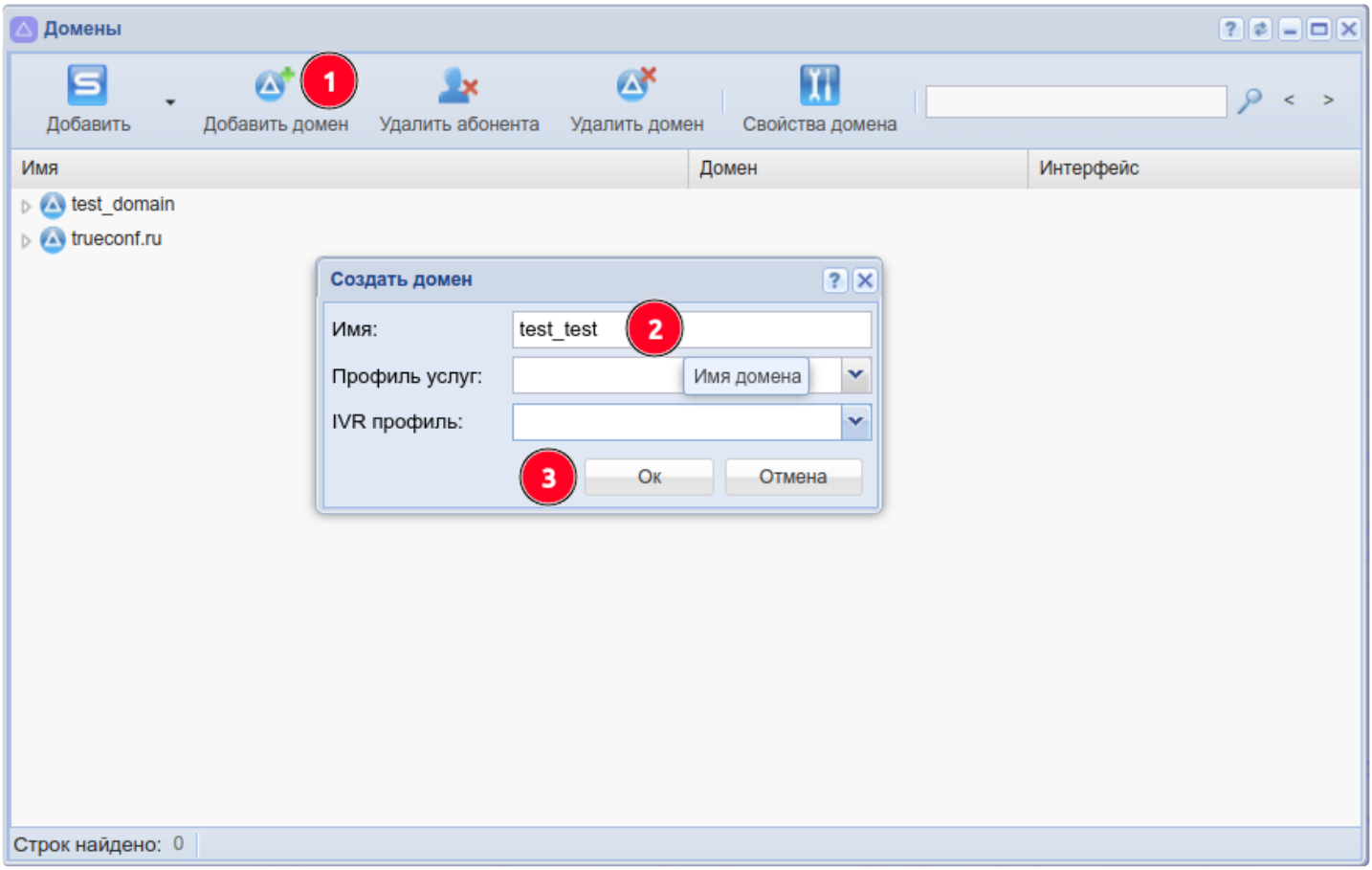
Пример создание через CLI (в примере имя домена "test_test"):

1. Создать домен с именем "test_domain":
   `/domain/declare test_test --add-domain-admin-privileges --add-domain-user-privileges`
2. Подключить к созданному домену, **ранее** созданный IP-set ("test_set"):
   `/domain/test_test/sip/network/set ip_set [test_set]`
3. Назначить на созданный домен доступные услуги (все или определенные/необходимые):
   `/cluster/storage/ds1/ss/access-list add test_domain *`
4. Установить на созданном домене политику применения лицензии с указанием имени (для примера "new_sub")
   `/domain/test_test/ss/licence/politics/declare new_sub`
5. Добавить на созданном домене в новую политику пакет лицензионных услуг (для примера "ECSS-FULL+")
   `/domain/test_test/ss/licence/politics/package-add new_sub ECSS-FULL+`
6. Активизируем новую политику применения лицензии на новом домене:
   `/domain/test_test/ss/licence/politics/activate new_sub`

Пример создание через Web (в примере имя домена "test_test") приложение "Домены":

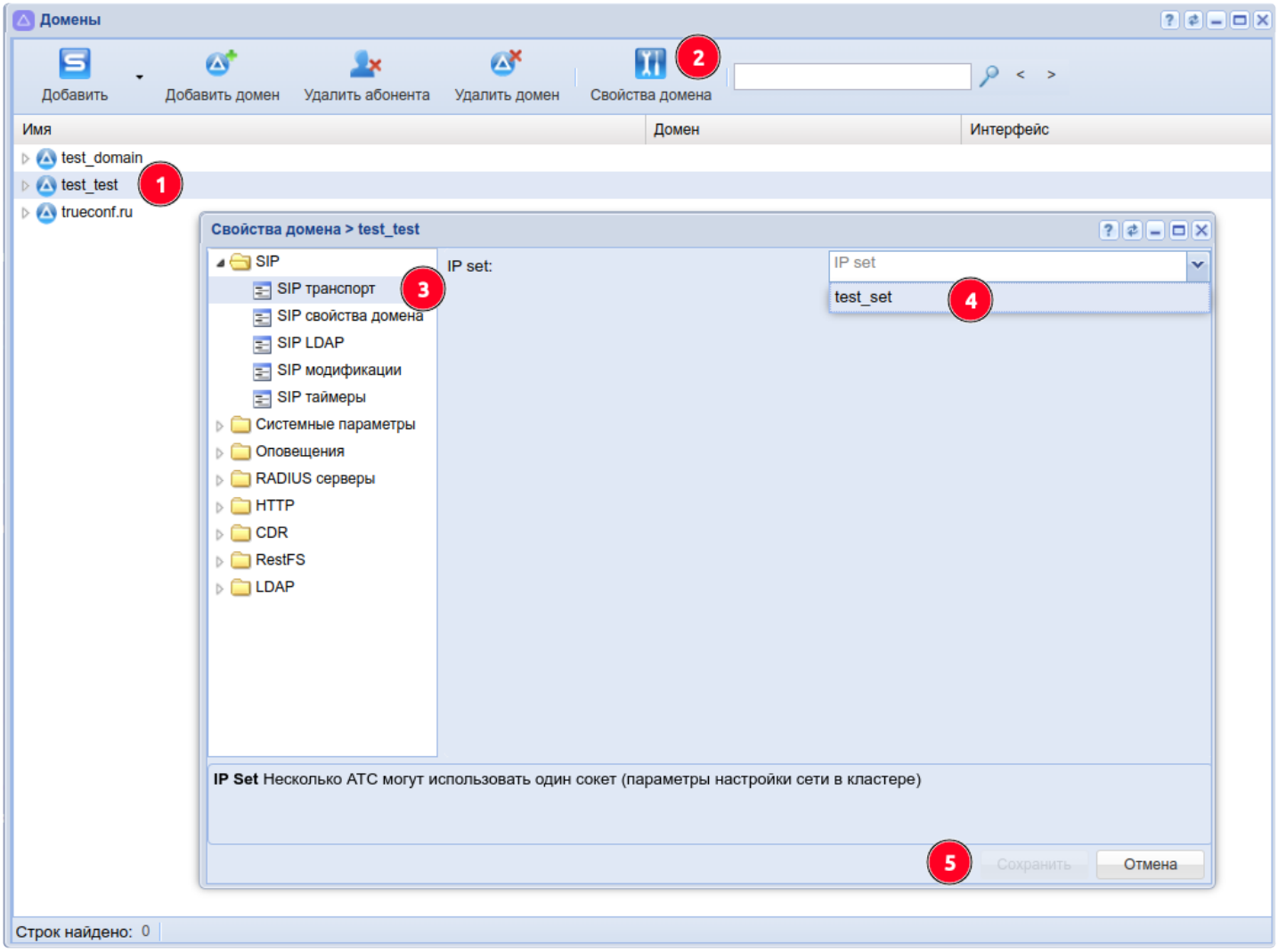
Создание домена

1. Выбрать "Добавить домен";
2. Указать имя нового домена (в примере "test_test");
3. Сохранить ("Ок").



Подключение IP set

1. Выбрать домен (в примере "test_test");
2. Выбрать "Свойства домена";
3. Выбрать "SIP/SIP транспорт";
4. Выбрать IP set, который будет обслуживать данный домен, из выпадающего списка;
5. Сохранить.



Конфигурацию услуг, для нового домена, выполнить в CLI.

После создания доменов в файловой системы автоматически создадутся нужные каталоги уровня домена.

Для примера :

/var/lib/ecss/modification/ctx/src/Имя_домена
/var/lib/ecss/routing/ctx/src/Имя_домена
/var/lib/ecss/cdИмя_домена

и т.д.

Настройка маршрутизации

Маршрутизация телефонных вызовов — это процесс определения интерфейса назначения для конкретного вызова на основании информации об интерфейсе источника вызова, информации о телефонном номере вызывающего и вызываемого абонента, категории вызывающего абонента, времени суток и дне недели.
Контекст маршрутизации — совокупность правил маршрутизации уникальная в домене маршрутизации, в рамках которого идет определение интерфейса вызываемого абонента.

Описание процесса маршрутизации вызовов в системе ECSS-10 приведено в разделе "Виртуальная АТС. Маршрутизация телефонных вызовов".

Создание контекстов маршрутизации

В соответствии с проектом необходимо создать контексты маршрутизации вызовов, которые в дальнейшем будут применяться в настройках доменов.

Это можно сделать несколькими способами:

- создать и импортировать контексты вручную в формате, указанном в документации. Контексты в формате xml создаются в каталоге **/var/lib/ecss/routing/ctx/src/<DOMAIN>**;
- создать и настроить контексты, используя приложение web-конфигуратора "Менеджер маршрутизации (Routing manager)" для каждого домена;
- опционально — настроить RADIUS-маршрутизацию.

Далее, если это определено в проекте, необходимо подготовить контексты модификации и адаптации также для каждого домена.

Применение контекстов маршрутизации для системных интерфейсов

Для интерфейсов **system:ivr** и **system:teleconference** необходимо назначить контексты маршрутизации. Настройки выполняются с помощью команд CLI. Описание и примеры команд приведены в разделе "/domain/<DOMAIN>/system-iface/ - команды управления системными интерфейсами".

Создание и настройка абонентов

В соответствии с проектом в доменах нужно создать требуемое количество абонентов. Сделать это можно с помощью команды CLI /domain/<DOMAIN>/sip/user/declare или приложения web-конфигуратора "Карточка абонента (Subscriber card)".

При создании абоненту нужно назначить номер, группу, контекст маршрутизации, CDR-группу, способ авторизации и авторизационные данные. Также сразу можно для каждого абонента назначить набор необходимых услуг и настроить необходимые ограничения.

Абоненты в домене — условие необязательное, в системе могут быть и чисто транзитные домены, на которых настраиваются соответствующие правила прохождения вызовов.

Создание и настройка транков

- **Транк** представляет собой совокупность ресурсов для обслуживания телефонных вызовов в заданном направлении (см. раздел "Транки и бриджи");
- **SIP-Транк** представляет собой направление, работающее по протоколу SIP/SIP-T/SIP-I;
- **Динамический транк** — транк с обязательной поддержкой регистрации. Для совершения вызова по динамическому транку взаимодействующий шлюз должен быть зарегистрирован по данному транку в системе ECSS-10.

Декларация и настройка транков производится:

- через интерфейс командной строки, см. раздел "/domain/<DOMAIN>/trunk/sip/ - команды управления транками SIP";
- через приложение web-конфигуратора "Менеджер транков (Trunk manager)".

Порядок создания и настройки транков приведен в разделе "Управление SIP-транками".

Далее на транках настраиваются необходимые сервисы.

Создание и настройка bridge-интерфейсов

Бридж — виртуальный транк, позволяющий соединять между собой две виртуальные АТС в рамках одной системы ECSS-10.

Если в системе имеется более одного домена, то связь между ними осуществляется с помощью **бриджей** (см. раздел "Транки и бриджи").

Бриджи создаются и настраиваются:

- через интерфейс командной строки, см. раздел "/bridge/ — команды управления bridge-интерфейсами";
- через приложение web-конфигуратора "Менеджер бриджей (Bridge manager)".

Настройка ограничений

Для каждого домена существует возможность задать разного рода ограничения в рамках лицензии.

Таблица 1. Список ограничений уровня домена

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 приведена 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.

Название свойства	Значение по умолчанию	Описание
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.
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.
elph\member\limi	infinity (ограничено лицензией)	Количество активных регистраций клиентов Elph.

⚠ Если у виртуальной АТС выставлено свойство vats_type transit или transit_offroad - вызовы не будут пропущены на IVR.

Ограничения настраиваются в соответствии с проектом:

- через интерфейс командной строки, см. раздел "/domain/<DOMAIN>/properties/restrictions/ - команды управления ограничениями виртуальной АТС";
- через приложение web-конфигуратора "Домены (Domains)" (Свойства домена -> Системные параметры -> Ограничения).

Сценарии IVR

Для каждой виртуальной АТС можно настроить дополнительные сценарии IVR, если они сразу предусмотрены проектом. Сценарии создаются в приложении web-конфигуратора "IVR-редактор (IVR editor)". Также управлять сценариями можно с помощью команд CLI. Описание и примеры приведены в разделе "/domain/<DOMAIN>/ivr/ - команды управления IVR-скриптами".

При необходимости для каждого домена можно настроить ограничения для работы IVR:

- через интерфейс командной строки, см. раздел "/system/ivr/script/restrictions/ - команды управления настройками ограничений IVR-скриптов";
- через приложение web-конфигуратора "Редактор IVR ограничений (IVR restrictions manager)".

17.4.4 Настройка дополнительных сервисов

Если предусмотрено проектом, на начальном этапе настраиваются также дополнительные сервисы.

Настройка взаимодействия с серверами RADIUS

Описание и настройка взаимодействия с подсистемой AAA (Authentication, Authorization, Accounting) приведены в разделе "Настройка динамических абонентов и системы RADIUS".

Порядок настройки взаимодействия с серверами AAA:

- настройка параметров виртуальной АТС для взаимодействия с сервером аутентификации/авторизации (RADIUS);
- настройка параметров виртуальной АТС для взаимодействия с сервером аккаунтинга (RADIUS);
- настройка ограничений связи при сбое сервера.

Настройка функции СОРМ

В комплексе ECSS-10 заложены возможности для выполнения требований к системе технических средств по обеспечению функций оперативно-розыскных мероприятий на электронных АТС, утвержденные приказом Госкомсвязи России от 20.04.1999 № 70 и приказом Минкомсвязи России №268 от 19.11.2012.

Порядок настройки функции СОРМ:

- проверка наличия соответствующей лицензии;
- настройка протокола взаимодействия с посредником СОРМ;
- настройка маршрутизации в соответствии с требованиями сотрудников спецслужб.

Описание и настройка системы СОРМ приведены в разделе "Система СОРМ" / " Система СОРМ3".

Настройка дополнительных приложений

В составе экосистемы ECSS-10 возможно использование дополнительных сервисных приложений, расширяющих функциональные возможности:

- Call-центр;
- инструментарий для проведения селекторных совещаний;
- сервис "Автообзвон";
- сервис автоматического распознавания речи (ASR);

- интеграции с Desktop-ассистент, CRM, Skype for business);
- сервис "Автосекретарь";
- визуализация статистических данных в системе мониторинга "Grafana";
- приложение "Портал абонента";
- система "Autoprovision (AUP)" для автоматического конфигурирования и обновления ПО телефонных аппаратов.

Описание данных приложений приведено в соответствующих разделах документации.

18 Рекомендации по обеспечению безопасности системы

- Общие требования
  - Telnet
  - Отключить доступ через root на сервер
  - Включить доступ только по ключам ssh
  - Использовать безопасные ключи rsa -b 4096
  - Автоматическая генерация паролей для SIP-пользователей
  - Изменение паролей пользователей
    - Изменение политики создания паролей пользователей "cocon"
    - Пользователь "admin"
      - В CoCon
      - В Web
    - Пользователи для операторов
    - Изменение пароля пользователя для клиента ecss-node
    - Изменение пароля пользователя для клиентов ecss-teleconference-ui и ecss-cc-ui
    - Изменение пароля пользователя для клиента ecss-restfs
    - Изменение паролей пользователей для клиентов ElphChatServer и ecss-call-history
    - Изменение пароля пользователя для клиента ecss-web-configurator
  - Использовать SBC, настроить firewall, ACL
  - Обновлять Ubuntu и утилиты сервера при выходе пакета безопасности
  - NTP
- Настройка безопасности OpenSSH
  - Настройка конфигурации сервера и клиента ssh
    - Обмен ключами
    - Аутентификация
    - Симметричные шифры
    - Код аутентичности сообщения (MAC)
    - Защита от утечки ключей
    - Ограничение пользователей
    - Ограничение интерфейсов
    - Смена порта

Рекомендации по безопасной настройке носят общий характер и подходят для большинства инсталляций. Настоящие рекомендации в значительной степени повышают безопасность эксплуатации устройства, но не являются исчерпывающими. В зависимости от схемы применения устройства необходимо настраивать и другие параметры безопасности. В некоторых специфических случаях выполнение данных рекомендаций может привести к неработоспособности сети. При настройке устройства стоит в первую очередь следовать техническим требованиям и регламентам сетей, в которых будет эксплуатироваться данное устройство.

Telnet - небезопасный протокол.
В Ubuntu действует политика "не открывать порты", которая распространяется на установку по умолчанию. Небезопасный протокол telnet **по умолчанию не включен** на сервере Ubuntu.
Не рекомендуется устанавливать telnet на серверах ECSS .

18.1 Общие требования

Для обеспечения необходимых настроек безопасности системы рекомендуется:

18.1.1 Telnet

⚠ В Ubuntu по умолчанию telnet выключен , так как не является безопасным протоколом. Не рекомендуется включать его.

18.1.2 Отключить доступ через root на сервер

По умолчанию учетная запись root отключена в Ubuntu, в отличии от других дистрибутивов Linux. Если root все-таки включен, отключить можно командой:

```
sudo passwd -l root
```

18.1.3 Включить доступ только по ключам ssh

В /etc/ssh/sshd_config:

```
PasswordAuthentication no
ChallengeResponseAuthentication no
```

18.1.4 Использовать безопасные ключи rsa -b 4096

```
sudo ssh-keygen -t rsa -b 4096 -f /etc/ssh/ssh_host_rsa_key
sudo ssh-keygen -o -a 100 -t ed25519 -f /etc/ssh/ssh_host_ed25519_key
```

18.1.5 Автоматическая генерация паролей для SIP-пользователей

Система ECSS-10 имеет возможность при создании SIP-абонентов генерировать пароли, которые соответствуют рекомендуемым требованиям безопасности.

Пример декларации абонента:

```
/domain/test_domain/sip/user/declare default_routing sip 42002 alias-as-user
qop_authentication common_login 42002 auto-generation
Executed on the sip1@ecss1
Intermediate (incomplete) result:
Declaration for range: 42002@test_domain..42002@test_domain (1)
...
1 interfaces check for existing ...
[**************************************************************] 70mks
1 users interfaces declaration ...
[**************************************************************] 25ms
1 users divided into 1 parts to declare ...
[**************************************************************] 60mks
```

```
1 users aliases declaration ...
[**************************************************************************] 4ms
1 users trying licence packages activating ...
[**************************************************************************] 2ms
8 interfaces recall to base
[**************************************************************************] 3ms
Executed on the sip1@ecss1
┌──────────────────────────┐
│ declared 1 subscribers │
└──────────────────────────┘
```

Просмотр учетных данных:

сгенерированный пароль --> VmaifqfMais5

```
/domain/test_domain/sip/user/info sip 42002@test_domain
. . .
┌──────────────────────┬──────────────────────────────────────────┐
│        User          │             42002@test_domain           │
├──────────────────────┼──────────────────────────────────────────┤
│internal iface name   │06d48c3938a30f5d                          │
│isActive              │false                                     │
│group                 │sip                                       │
│authentication        │42002:VmaifqfMais5 (ds) use qop: true     │
. . .
```

18.1.6 Изменение паролей пользователей

Изменение политики создания паролей пользователей "cocon"

Отредактировать файл (нас интересует выделенная секция)

/etc/ecss/global.config

```
%% File autogenerate.
%% For change, remove it and run: dpkg-reconfigure
[
    {ecss_env, [
        {config_root,   "/etc/ecss"},
        {data_root,     "/var/lib/ecss"},
        {log_root,      "/var/log/ecss"}
        ]},
    {rps, [
        {thresholds, [
            {disk, [
                {enabled, true},
                {points, ["/var/lib/mysql"]}
```

```
                ]}
            ]},
        {monitoring, [
            %{applications, [%%APPS_MON%%]}
            %{interfaces, [{noAlarmFor, [%%IFACES_MON%%]}]}
            ]}
        ]},
    {chronica, [
        {tty_enabled, true} %applyed for journald
        ]},
    {cocon, [
        % ms, delay for node in cluster group
        {heartbeat, 3000},
        % name of amqp exchange
        {exchange, "ccn.direct"},
        % Type DB storage for user authentication (ldap, mysql, undefined)
        %% default: mysql
        {user_data, 'mysql'},
        % Type DB for audit data (ldap, mysql, undefined)
        %% default: mysql
        {audit, 'mysql'},

        %%%% Settings LDAP (not required)
        %% ldap ip
        {ldap_server_ip, "cocon.ldap.ecss"},
        %% ldap port
        {ldap_server_port, "389"},
        %% DN for storage users
        {ldap_people_dn, "ou=people,dc=ecss"},
        %% DN for storage groups
        {ldap_groups_dn, "ou=groups,dc=ecss"},
        %% DN for storage events
        {ldap_audit_dn, "ou=audit,dc=ecss"},
        %% DN for ldap usages (audit, etc)
        {ldap_system_username, "cn=admin,dc=ecss"},
        {ldap_system_password, "admin"},

        %%%% Settings MySQL (not required)
        %% MySQL ip
        {mysql_server_ip, "cocon.mysql.ecss"},
```

```
        %% MySQL port
        %% default: undefined
        {mysql_server_port, "3306"},
        {mysql_username, "audit"},
        {mysql_password, "audit"},
        % Usage BD
        {mysql_db, "ecss_audit"},
        % Time (in UTC) for every day cleaning audit data
        {audit_clean_time, {2, 0, 0}},
        % Count of date for save audit data
        {audit_save_days, 90},
        {mount_cmd, "/etc/ecss/ssh_command"},
        %% Password expire lifetime (infinity or days count)
        {password_lifetime, infinity},
        %% Allow reuse of old CoCon passwords
        {allow_passwords_reuse, true},
        %% Maximum size of password history
        {password_history_depth, 10},
        % Security restrictions for CoCon's users' password
        {password_restrictions, [
            % Minimum password length (digit or unlimited)
            {min_length, unlimited},
            % Maximum password length (digit or unlimited)
            {max_length, unlimited},
            % Digits symbol required [0..9]
            {digits_required, false},
            % Special characters required: ( !"#$%&'()*+,-./:;<=>?@[\]^_`{|}~)
            {special_characters_required, false},
            % Latin letter at uppercase special_characters_required [A..Z]
            {uppercase_letters_required, false},
            % Latin letter at lowercase special_characters_required [a..z]
            {lowercase_letters_required, false},
            % National (other unicode) characters allowed or not
            {national_characters_not_allowed, false},
            % List of prohibited passwords
            {banned_passwords, []}
         ]}
        ]},
```

```
    {ecss_mgmt, [
        {os_cmd, [
            {host_shell, "trap 'echo INTERRUPTED' SIGINT; ssh localhost"}
        ]}
    ]}
].
```

По умолчанию значение параметров = unlimited/false. Нужно их установить / активировать.

Установим новые значения

```
...
        % Security restrictions for CoCon's users' password
        {password_restrictions, [
            % Minimum password length (digit or unlimited)
            {min_length, 8},
            % Maximum password length (digit or unlimited)
            {max_length, 14},
            % Digits symbol required [0..9]
            {digits_required, true},
            % Special characters required: ( !"#$%&'()*+,-./:;<=>?@[\]^_`{|}~)
            {special_characters_required,true},
            % Latin letter at uppercase special_characters_required [A..Z]
            {uppercase_letters_required,true},
            % Latin letter at lowercase special_characters_required [a..z]
            {lowercase_letters_required,true},
            % National (other unicode) characters allowed or not
            {national_characters_not_allowed, false},
            % List of prohibited passwords
            {banned_passwords, []}
         ]}
...
```

Файл /etc/ecss/global.config нужно отредактировать на обоих хостах, и информация в файлах должна быть идентична.

Этим мы указали использовать в пароле не менее 8 символов, включающих в себя строчные и заглавные буквы латинского алфавита, а также цифры и спецсимволы. Так же можно добавить буквы Кириллицы и запретить использовать простые пароли (для примера Qwerty-1).

Чтобы изменения вступили в силу нужно выключить , а затем включить **все** ноды. Выполняется последовательно , вначале на ecss1 затем на ecss2. Для этого выполнить следующие команды:

выключение

```
sudo systemctl stop ecss-mediator.service
sudo systemctl stop ecss-pa-sip.service
sudo systemctl stop ecss-core.service
sudo systemctl stop ecss-ds.service
sudo systemctl stop ecss-mycelium.service
```

включение

```
sudo systemctl start ecss-mycelium.service
sudo systemctl start ecss-ds.service
sudo systemctl start ecss-core.service
sudo systemctl start ecss-pa-sip.service
sudo systemctl start ecss-mediator.service
```

Пользователь "admin"

По умолчанию после инсталляции ECSS для доступа в CLI и web-конфигуратор автоматически создается пользователь **admin** со стандартным паролем **password** и пользователь **support** предназначенный для работы специалистов техподдержки .

Рекомендуется заменить дефолтный пароль "password" на более безопасный. Это можно выполнить как в CoCon, так и в Web интерфейсе.

В CoCon

выполняется в CoCon

```
/cocon/passwd admin
```

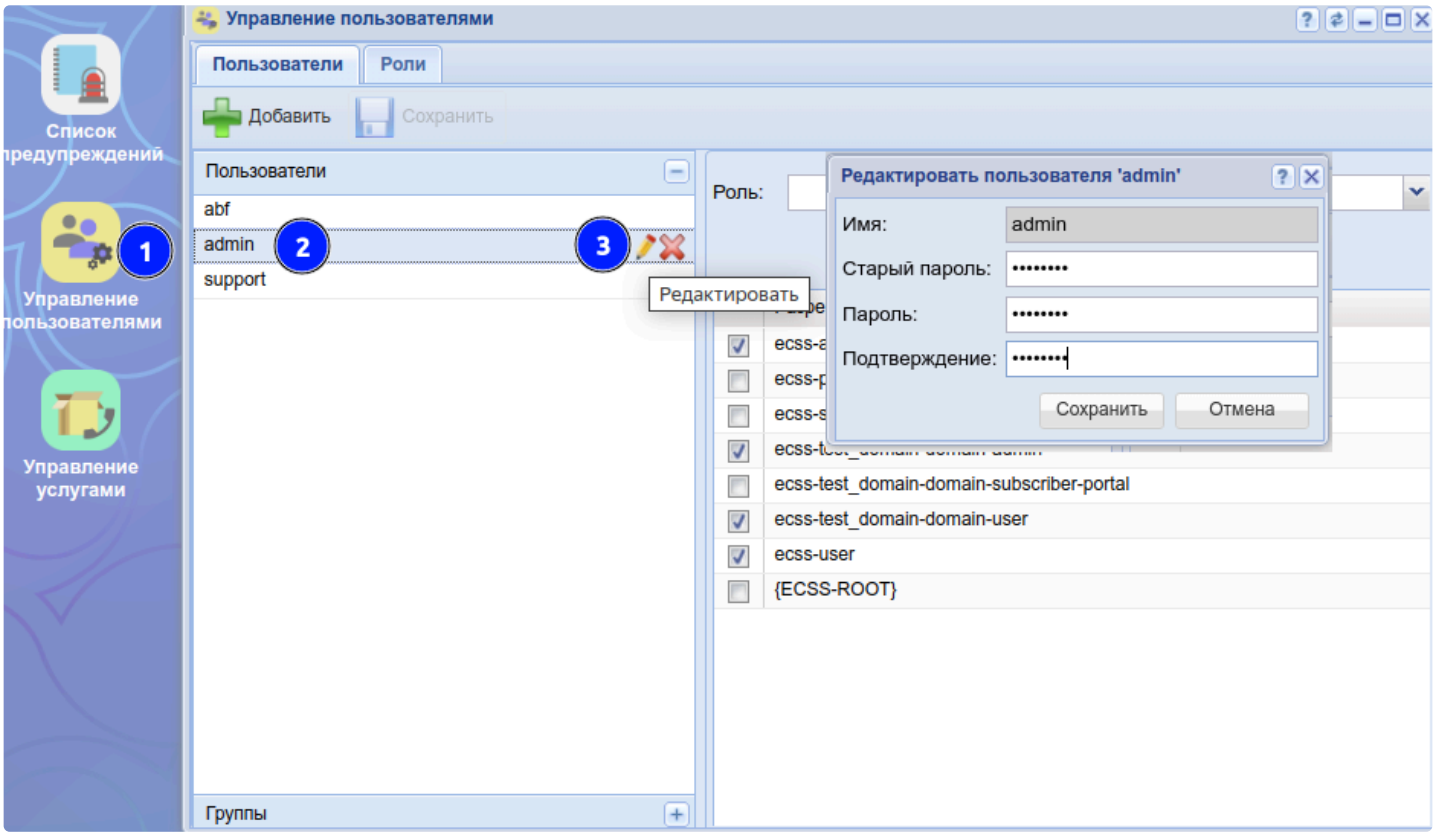
Пример:

```
cocon/passwd admin
[passwd] Changing password for admin
Enter new password:
Confirm new password:
Password for user "admin" has been successfully changed
```

В Web

1. Открыть приложение "Управление пользователями"
2. Выбрать пользователя "admin"

3. Выбрать "Редактировать"
4. В окне "Редактировать пользователя "admin"", указать Старый пароль/Новый пароль/Подтвердить новый пароль.
5. Сохранить



Пользователи для операторов

Нужно создать новых пользователей с правами администратора системы, для каждого оператора. Пароль данных пользователей, должен исключить возможность подбора. Для генерации сложных паролей можно воспользоваться стандартной утилитой Linux — **pwgen**. Выполенить это можно как через Cocon, так и через web:

Добавить пользователя через CoCon

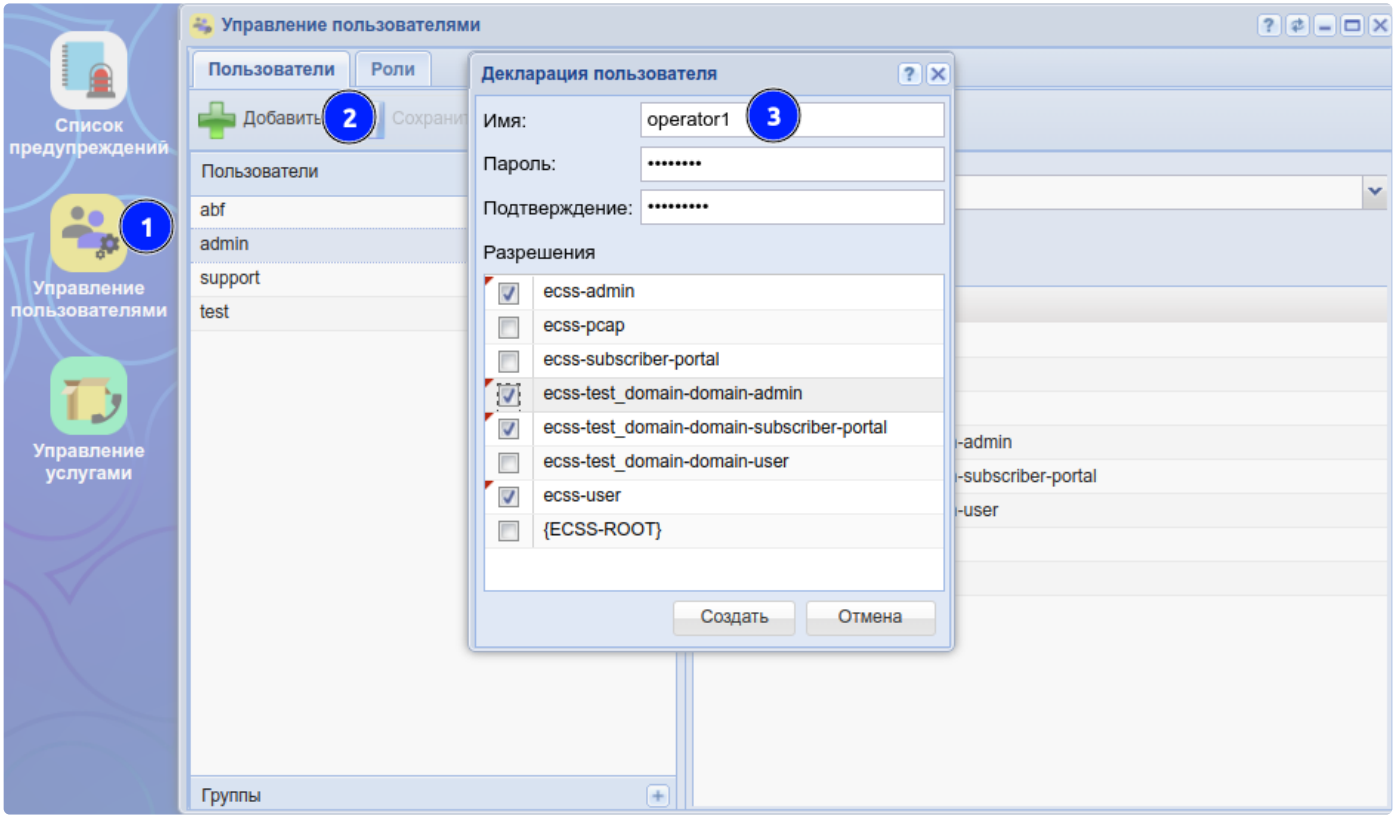
```
/cocon/add-user test
Enter password:
Confirm password:
ECSS global user: [n]/y ?> y
ECSS global administrator: [n]/y ?> y
User test has been successfully created
```

```
/cocon/list users
┌─────────┬──────────────────────────────────────────────────┬─────┐
│ Login   │                     Groups                       │Roles│
├─────────┼──────────────────────────────────────────────────┼─────┤
│test     │ecss-admin                                        │     │
│         │ecss-test_domain-domain-admin                     │     │
```

	ecss-test_domain-domain-user	
	ecss-user	
support	ecss-admin	
	ecss-user	
admin	ecss-admin	
	ecss-test_domain-domain-admin	
	ecss-test_domain-domain-user	
	ecss-user	
abf	ecss-admin	
	ecss-pcap	
	ecss-subscriber-portal	
	ecss-test_domain-domain-admin	
	ecss-test_domain-domain-subscriber-portal	
	ecss-test_domain-domain-user	
	ecss-user	

Добавить пользователя через Web

1. Открыть приложение "Управление пользователями";
2. Выбрать "Добавить";
3. Указать Имя нового пользователя/Пароль нового пользователя/Подтвердить пароль;
4. Выбрать какие разрешения предоставить новому пользователю и для каких доменов;
5. Создать;
6. Сохранить.



Изменение пароля пользователя для клиента ecss-node

Сервис **ecss-node** использует для подключения к БД MySql пользователя "**audit**". Для его модификации требуется отредактировать файл **/etc/ecss/global.config** (секция cocon, параметр mysql_password). После модификации, для того чтобы изменения вступили в силу, потребуется перезагрузка сервисов **ecss-mycelium.service**, **ecss-mediator.service**, **ecss-core.service**, **ecss-ds.service**.

на обоих хостах

```
sudo nano /etc/ecss/global.config
```

для примера меняем пароль "audit" на "audit!1"

```
. . .
        %%%% Settings MySQL (not required)
        %% MySQL ip
        {mysql_server_ip, "cocon.mysql.ecss"},
        %% MySQL port
        %% default: undefined
        {mysql_server_port, "3306"},
        {mysql_username, "audit"},
```

```
        {mysql_password, "audit!1"},
        % Usage BD
        {mysql_db, "ecss_audit"},
 . . .
```

на обоих хостах

```
sudo systemctl restart ecss-mediator.service
```

на обоих хостах

```
sudo systemctl restart ecss-mycelium.service
```

на обоих хостах

```
sudo systemctl restart ecss-core.service
```

на обоих хостах

```
sudo systemctl restart ecss-ds.service
```

Изменение пароля пользователя для клиентов ecss-teleconference-ui и ecss-cc-ui

Эти сервисы используют для подключения к ДБ Postgres пользователя "**postgres**". Для модификации пароля необходимо выполнить команду **dpkg-reconfigure** для сервиса ecss-teleconference-ui /ecss-cc-ui соответственно.

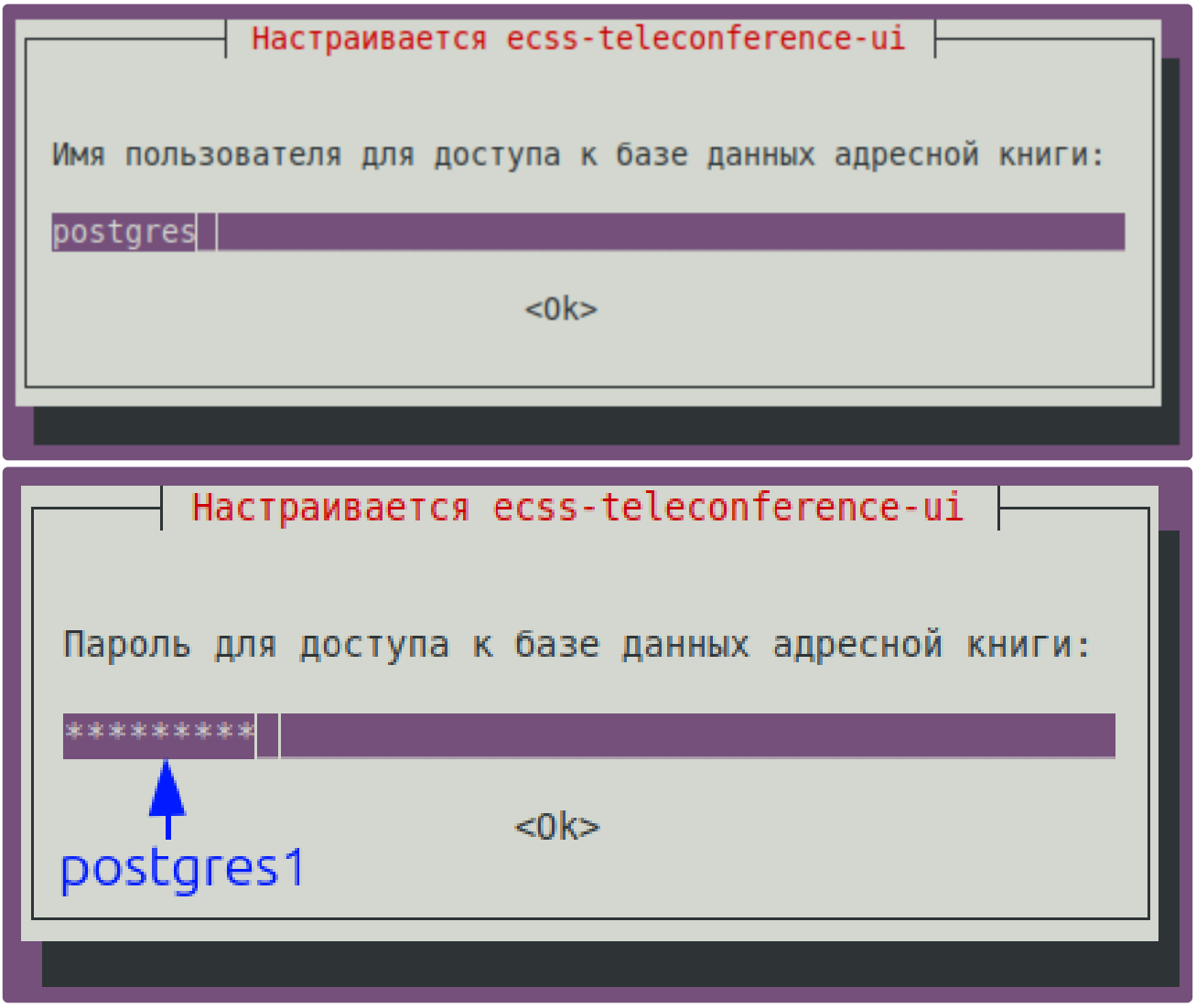
на обоих хостах

```
sudo dpkg-reconfigure ecss-teleconference-ui
```

на обоих хостах

```
sudo dpkg-reconfigure ecss-cc-ui
```

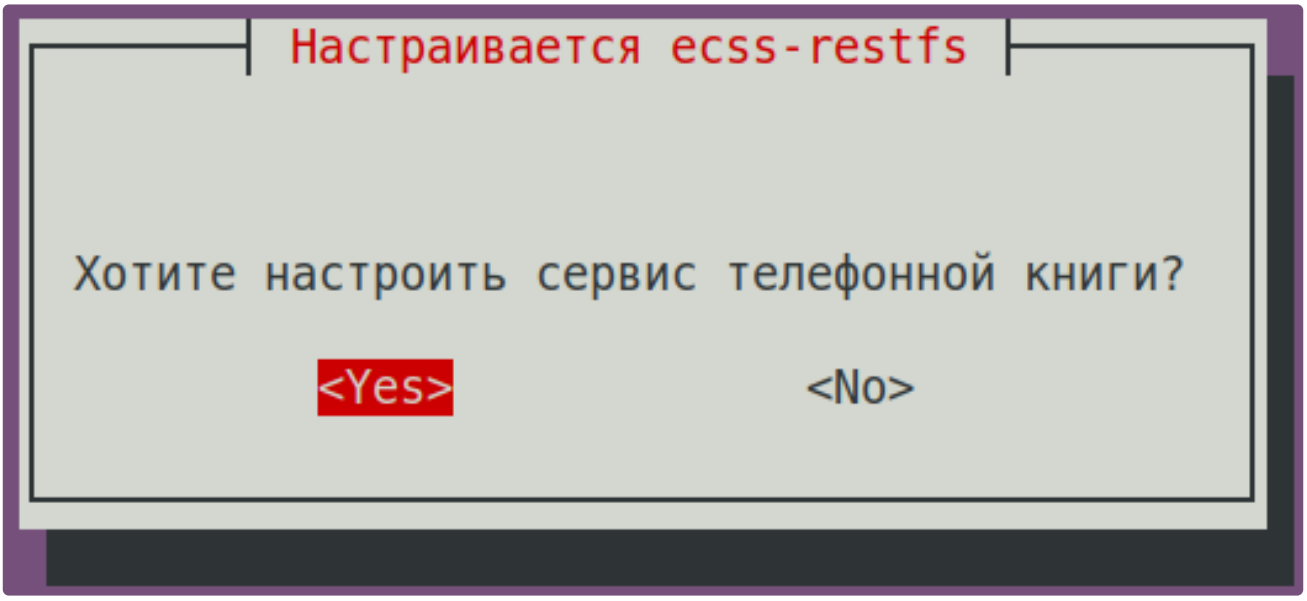
В процессе конфигурации будет задан вопрос о пароле для пользователя **address_book** - необходимо указать новый пароль (в примере сервис ecss-teleconference-ui).



Изменение пароля пользователя для клиента ecss-restfs

Сервис **ecss-restfs** использует для подключения к ДБ Postgres пользователя "**postgres**". Для модификации пароля необходимо выполнить команду dpkg-reconfigure ecss-restfs.

В процессе конфигурации будет заданы вопросы:



Изменение паролей пользователей для клиентов ElphChatServer и ecss-call-history

Эти сервисы используют для подключения к ДБ MySql пользователей "**address_book**", "**ecss_meeting_db**", "**calls_db**". Для модификации пароля необходимо выполнить модификацию конфигурационного файла. Для сервиса ElphChatServer - /opt/elph-chat-server/stable/.elph-chat.env. Для сервиса ecss-call-history / opt/ecss-call-history/stable/.env.

на сервере Chat Server/Call History

```
sudo nano /opt/elph-chat-server/stable/.elph-chat.env
```

изменить пароль на новый

```
. . .
# Параметры подключения к ecss-mysql (БД meeting-db - для работы чатов телеконференций)
ECSS_MEETING_DB_HOST="${ELPH_SSW}"
ECSS_MEETING_DB_PORT=3306
ECSS_MEETING_DB_NAME=ecss_meeting_db
ECSS_MEETING_DB_USER=ecss_meeting_db
ECSS_MEETING_DB_PASS=ecss_meeting_db#3
```

```
ECSS_ADDRESSBOOK_DB_NAME=ecss_address_book
ECSS_ADDRESSBOOK_DB_PASSWORD=address_book@2
ECSS_ADDRESSBOOK_DB_USER=address_book
. . .
```

на сервере Chat Server/Call History

```
sudo nano /opt/ecss-call-history/unstable/.env
```

изменить пароль на новый

```
. . .
#######################################################
# External endpoints
#######################################################
# ecss-meeting DB
ECSS_MEETING_DB_HOST="192.168.107.108"
ECSS_MEETING_DB_PORT=3306
ECSS_MEETING_DB_NAME=ecss_meeting_db
ECSS_MEETING_DB_USER=ecss_meeting_db
ECSS_MEETING_DB_PASS=ecss_meeting_db%5
# ecss-calls DB
ECSS_CALLS_DB_HOST="192.168.107.108"
ECSS_CALLS_DB_PORT=3306
ECSS_CALLS_DB_NAME=ecss_calls_db
ECSS_CALLS_DB_USER=calls_db
ECSS_CALLS_DB_PASS=calls_db^6
ECSS_CALLS_DB_TIMEZONE=+07:00
# ecss-addressbook DB
ECSS_ADDRESS_BOOK_DB_HOST="192.168.107.108"
ECSS_ADDRESS_BOOK_DB_PORT=3306
ECSS_ADDRESS_BOOK_DB_NAME=ecss_address_book
ECSS_ADDRESS_BOOK_DB_USER=address_book
ECSS_ADDRESS_BOOK_DB_PASS=address_book@2
#######################################################
```

```
. . .
```

После изменения паролей требуется обновить сервис ElphChatServer и ecss-call-history

ElphChatServer

```
/opt/elph-chat-server/stable/.ecss-elph-chat-server.sh up
```

ecss-call-history

```
/opt/elph-chat-server/stable/.ecss-call-history.sh up
```

Изменение пароля пользователя для клиента ecss-web-configurator

Этот сервис используют для подключения к ДБ MySql пользователя "**web_conf**". Для модификации пароля необходимо выполнить модификацию файла **/usr/share/ecss/ecss-web-conf/settings.py**.

/usr/share/ecss/ecss-web-conf/settings.py

```
sudo nano /usr/share/ecss/ecss-web-conf/settings.py
```

/usr/share/ecss/ecss-web-conf/settings.py

```
. . .
_DEB_DB_NAME = 'ecss_web_conf'
_DEB_USER = 'postgres'
_DEB_PASS = 'postgres1'
_DEB_HOST = '127.0.0.1'
_DEB_PORT = '5439'
_DEB_POOL_SIZE = '10'
. . .
```

Для применения изменений выполнить команду:

```
sudo dpkg-reconfigure ecss-web-conf
```

18.1.7 Использовать SBC, настроить firewall, ACL

Рекомендуется настроить свой firewall таким образом, чтобы Restfs был доступен только с самого сервера и недоступен внешне.

Если сервер или SBC имеет прямой доступ в интернет или корпоративную сеть, где не исключены разного рода попытки неавторизованных запросов или DDoS-атаки, нужно настроить список разрешенных IP или подсетей, с которых разрешен доступ к определенным портам. Для остальных доступ должен быт запрещен. Если в составе комплекса есть SBC, то в нем уже реализован необходимый функционал для защиты от внешних атак — динамический брандмауэр с механизмом распознавания атак и попыток несанкционированного доступа. В SBC также можно настроить черные/белые списки подсетей для доступа как по управлению, так и по SIP.

18.1.8 Обновлять Ubuntu и утилиты сервера при выходе пакета безопасности

Рекомендуется не пренебрегать периодическими обновлениями системы, особенно при выпуске свежих версий пакетов, улучшающих функционал безопасности.

18.1.9 NTP

Рекомендуется всегда настраивать синхронизацию системных часов с доверенными источниками сетевого времени (NTP). Алгоритм настройки NTP приведён в разделе установка ecss-node входит в стандартную установку пакетов ECSS.

18.2 Настройка безопасности OpenSSH

На серверах ECSS должны быть установлены пакеты openssh-client openssh-server или мета-пакет ssh, включающий в себя клиента и сервер.

18.2.1 Настройка конфигурации сервера и клиента ssh

Конфигурация сервера OpenSSH описана в файле /etc/ssh/sshd_config. Для более полного обеспечения безопасности необходимо изменить некоторые настройки, принятые по умолчанию.

В репозитории Ubuntu-18.04 LTS текущая версия OpenSSH_7.6p1. Нижеприведенные рекомендации по конфигурации для версий, начиная с 7.6.

В OpenSSH 7.0 и старше отключена поддержка протокола SSH 1.0 по умолчанию во время компиляции. Точно так же по умолчанию во время компиляции больше не активируется поддержка ключей `diffie-hellman-group1-sha1` размером 1024 бит, хостов и пользовательских ключей `ssh-dss`, `ssh-dss-cert-*`. Не поддерживается формат сертификатов v00, а опция `PermitRootLogin` по умолчанию изменила значение с `yes` на `prohibit-password`.

Обмен ключами

Для использования только заслуживающих доверия протоколов обмена ключами в /etc/ssh/sshd_config для сервера следует указать:

```
KexAlgorithms curve25519-sha256@libssh.org,diffie-hellman-group-exchange-sha256
```

Аналогичные настройки для клиента, в /etc/ssh/ssh_config:

```
Host *
    KexAlgorithms curve25519-sha256@libssh.org,diffie-hellman-group-exchange-sha25
```

Аутентификация

В SSH поддерживается четыре алгоритма аутентификации по открытым ключам: DSA, ECDSA, Ed25519 и RSA.

ECDSA завязан на технологиях NIST и должен быть отключен. Так как размер ключей DSA не может превышать 1024 бит, его тоже следует отключить. Поэтому рекомендуется удалить все ключи:

```
cd /etc/ssh
rm ssh_host_*key*
```

Далее следует сгенерировать ключ RSA большего размера, а также добавить ключ ed25519:

```
sudo ssh-keygen -t rsa -b 4096 -f /etc/ssh/ssh_host_rsa_key
sudo ssh-keygen -o -a 100 -t ed25519 -f /etc/ssh/ssh_host_ed25519_key
```

Симметричные шифры

Из 15 поддерживаемых в SSH алгоритмов симметричного шифрования, используемых для организации защиты установленного канала связи, безопасными можно считать chacha20-poly1305, aes*-ctr и aes*-gcm. Шифры 3des-cbc и arcfour потенциально уязвимы в силу использования DES и RC4, cast128-cbc применяет слишком короткий размер блока (64 бит).

Поэтому в /etc/ssh/sshd_config рекомендуется добавить:

```
Ciphers aes256-gcm@openssh.com,aes128-gcm@openssh.com,chacha20-
poly1305@openssh.com,aes256-ctr,aes192-ctr,aes128-ctr
```

В /etc/ssh/ssh_config:

```
Host *
    Ciphers aes256-gcm@openssh.com,aes128-gcm@openssh.com,chacha20-
poly1305@openssh.com,aes256-ctr,aes192-ctr,aes128-ctr
```

Код аутентичности сообщения (MAC)

Для шифров в режиме CTR для гарантирования целостности передаваемых блоков доверия заслуживает только метод Encrypt-then-MAC ("*-etm", MAC добавляется к уже зашифрованному блоку). Методы MAC-then-encrypt и Encrypt-and-MAC потенциально подвержены атакам. Из 18 доступных в SSH алгоритмов MAC сразу следует отбросить основанные на хэшах MD5 и SHA1, не стойких к выявлению коллизий, а также алгоритмы, использующие размеры ключей менее 128 бит и размеры тегов менее 256 бит. В итоге, наиболее безопасными MAC можно считать hmac-sha2-512-etm и hmac-sha2-256-etm.

В /etc/ssh/sshd_config:

```
MACs hmac-sha2-512-etm@openssh.com,hmac-sha2-256-etm@openssh.com
```

В /etc/ssh/ssh_config:

```
Host *
MACs hmac-sha2-512-etm@openssh.com,hmac-sha2-256-etm@openssh.com
```

Защита от утечки ключей

Наиболее простым способом получения контроля за SSH-соединением является захват ключей на стороне клиента или сервера. Рекомендации сводятся к соблюдению типовых правил поддержания безопасности системы:

- оперативная установка обновлений, установка программ только из надёжных источников;
- установка только действительно необходимых программ и сервисов;
- использование программ, для которых доступны исходные тексты, включение дополнительных механизмов защиты (Grsecurity, сборка с флагом -fstack-protector).

Для защиты ключей следует выбрать надёжный пароль доступа к клиентским файлам ключей. При формировании ключа для увеличения числа итераций хэширования можно использовать опцию "ssh-keygen -o -a число", что усложнит подбор пароля. Также можно сохранить ключи только на внешнем носителе, подключая его только во время соединения по SSH.

Ограничение пользователей

Добавляем параметр AllowUsers, которого нет в конфигурационном файле по умолчанию. Этот параметр разрешает доступ к серверу по протоколу SSH только для перечисленных пользователей.

В /etc/ssh/sshd_config:

```
AllowUsers dmitry support
```

Ограничение интерфейсов

Эти строки отвечают за настройку разграничений по сетевым интерфейсам, сетевому адресу или имени компьютера. По умолчанию сервер «слушает» (принимает подключения) на всех сетевых интерфейсах. Если нужно оставить подключение только через определенные интерфейсы, то раскомментировать строку:

```
ListenAddress 192.168.1.21
ListenAddress 10.16.33.5
```

В этой же строке можно явно указать порт, предварительно закомментировав (поставив символ # в начале строки)

#ListenAddress 0.0.0.0

Смена порта

Если есть вероятность попыток подбора пароля (например сервер в публичной сети) можно изменить порт на другой.

```
Port 22025
```

По умолчанию используется 22 порт. Изменим его на нестандартный порт 22025 — это избавит наш сервер от сетевых роботов, которые автоматически сканируют интернет в поиске открытых портов и пытаются через них подключиться. В основном, боты настроены на поиск стандартных портов. Это не избавит от сканирования человеком, но для защиты от человека существует файервол, хитрые способы открытия порта и пр.

Примерный листинг получившихся файлов конфигурации:

/etc/ssh/sshd_config:

```
#    $OpenBSD: sshd_config,v 1.101 2017/03/14 07:19:07 djm Exp $

# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

Port 22025
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::
ListenAddress 192.168.1.21
ListenAddress 10.16.33.5

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

```
#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile    .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
#PermitEmptyPasswords no

# Change to yes to enable challenge-response passwords (beware issues with
# some PAM modules and threads)
ChallengeResponseAuthentication no

# Kerberos options
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
#KerberosGetAFSToken no

# GSSAPI options
#GSSAPIAuthentication no
#GSSAPICleanupCredentials yes
#GSSAPIStrictAcceptorCheck yes
#GSSAPIKeyExchange no

# Set this to 'yes' to enable PAM authentication, account processing,
# and session processing. If this is enabled, PAM authentication will
# be allowed through the ChallengeResponseAuthentication and
# PasswordAuthentication.  Depending on your PAM configuration,
# PAM authentication via ChallengeResponseAuthentication may bypass
# the setting of "PermitRootLogin without-password".
# If you just want the PAM account and session checks to run without
# PAM authentication, then enable this but set PasswordAuthentication
# and ChallengeResponseAuthentication to 'no'.
UsePAM yes

#AllowAgentForwarding yes
#AllowTcpForwarding yes
#GatewayPorts no
X11Forwarding yes
#X11DisplayOffset 10
#X11UseLocalhost yes
#PermitTTY yes
PrintMotd no
#PrintLastLog yes
#TCPKeepAlive yes
```

```
#UseLogin no
#PermitUserEnvironment no
#Compression delayed
#ClientAliveInterval 0
#ClientAliveCountMax 3
#UseDNS no
#PidFile /var/run/sshd.pid
#MaxStartups 10:30:100
#PermitTunnel no
#ChrootDirectory none
#VersionAddendum none

# no default banner path
#Banner none

# Allow client to pass locale environment variables
AcceptEnv LANG LC_*

# override default of no subsystems
Subsystem    sftp    /usr/lib/openssh/sftp-server

# Example of overriding settings on a per-user basis
#Match User anoncvs
#    X11Forwarding no
#    AllowTcpForwarding no
#    PermitTTY no
#    ForceCommand cvs server

# @override for ECSS

KexAlgorithms curve25519-sha256@libssh.org,diffie-hellman-group-exchange-sha256
Ciphers aes256-gcm@openssh.com,aes128-gcm@openssh.com,chacha20-poly1305@openssh.com,aes256-c
tr,aes192-ctr,aes128-ctr
MACs hmac-sha2-512-etm@openssh.com,hmac-sha2-256-etm@openssh.com

HostKey /etc/ssh/ssh_host_ed25519_key
HostKey /etc/ssh/ssh_host_rsa_key
```

/etc/ssh/ssh_config:

```
# This is the ssh client system-wide configuration file. See
# ssh_config(5) for more information. This file provides defaults for
# users, and the values can be changed in per-user configuration files
# or on the command line.

# Configuration data is parsed as follows:
# 1. command line options
# 2. user-specific file
# 3. system-wide file
# Any configuration value is only changed the first time it is set.
# Thus, host-specific definitions should be at the beginning of the
# configuration file, and defaults at the end.

# Site-wide defaults for some commonly used options. For a comprehensive
# list of available options, their meanings and defaults, please see the
# ssh_config(5) man page.
```

```
Host *
# ForwardAgent no
# ForwardX11 no
# ForwardX11Trusted yes
# PasswordAuthentication yes
# HostbasedAuthentication no
# GSSAPIAuthentication no
# GSSAPIDelegateCredentials no
# GSSAPIKeyExchange no
# GSSAPITrustDNS no
# BatchMode no
# CheckHostIP yes
# AddressFamily any
# ConnectTimeout 0
# StrictHostKeyChecking ask
# IdentityFile ~/.ssh/id_rsa
# IdentityFile ~/.ssh/id_dsa
# IdentityFile ~/.ssh/id_ecdsa
# IdentityFile ~/.ssh/id_ed25519
# Port 22
# Protocol 2
# Ciphers aes128-ctr,aes192-ctr,aes256-ctr,aes128-cbc,3des-cbc
# MACs hmac-md5,hmac-sha1,umac-64@openssh.com
# EscapeChar ~
# Tunnel no
# TunnelDevice any:any
# PermitLocalCommand no
# VisualHostKey no
# ProxyCommand ssh -q -W %h:%p gateway.example.com
# RekeyLimit 1G 1h
 SendEnv LANG LC_*
 HashKnownHosts yes
 GSSAPIAuthentication yes
# @override for ECSS
 Ciphers aes256-gcm@openssh.com,aes128-gcm@openssh.com,chacha20-poly1305@openssh.com,aes256-c
tr,aes192-ctr,aes128-ctr
 KexAlgorithms curve25519-sha256@libssh.org,diffie-hellman-group-exchange-sha256
 MACs hmac-sha2-512-etm@openssh.com,hmac-sha2-256-etm@openssh.com
```

Проверить синтаксис файла конфигурации можно командой:

```
sshd -t
```

Посмотреть текущие настройки сервера:

```
sshd -T
```

19 Интеграция SSW и ELM сервером

- Типы лицензирования
- Установка лицензирования
  - Установка default лицензирования
  - Установка distributed лицензирования

19.1 Типы лицензирования

Начиная с версии ECSS10 3.14.15.4 поддерживается два типа лицензирования:

Причины перехода на версию "distributed" следующие:

- Позволяет использовать лицензии без Token в системе.
- Позволяет устанавливать SSW на хосты с именами отличными от ecss1 / ecss2.

default

Лицензирование по умолчанию. Для работы нужно указать паспорт (идентификатор_проекта.passport) / лицензию (идентификатор_проекта.lic), а также организовать доступ к токену.
В таком случае, после установки SSW будет работать как и раньше.

> Система может работать и в "старом режиме" когда используется только паспорт / лицензия. В таком случае, установка **ecss-elm-adapter** не нужна.

distributed

Лицензирование с помощью сервиса ELM.
При регистрации лицензии в Eltex формируется Product-Id и License-key с помощью которых можно будет получить лицензию уже из сервиса ELM.
Для работы elm лицензирования, необходимо установить сервис **ecss-elm-adapter** который реализует интеграцию с ELM .

19.2 Установка лицензирования

19.2.1 Установка default лицензирования

Чтобы узлы системы смогли встать в работу, надо сконфигурировать систему указав имена хостов на которых развёрнуты **ecss-сервисы** для примера ecss1/ecss2.

выполняется в CoCon

```
/system/clusters/set [ecss1, ecss2]
```

выполнить перезагрузку сервисов ECSS

на обоих хостах

```
sudo systemctl restart ecss-mycelium ecss-ds ecss-core ecss-pa-sip ecss-mediator
```

Далее необходимо загрузить паспорт и лицензии в систему:

```
cluster/storage/ds1/licence/set-passport <ssw passport>
ok
cluster/storage/ds1/licence/add <ssw licence>
ok
```

После установки хостов будет совершена повторная синхронизация кластера и можно будет запускать сервисы **ecss-mediator / ecss-core / ecss-pa-sip**

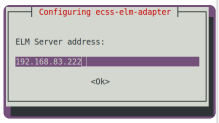
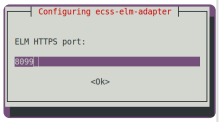
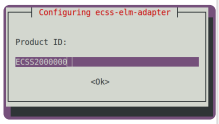
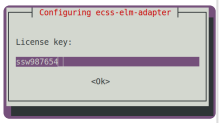
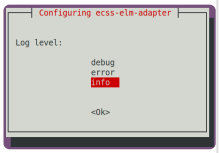
19.2.2 Установка distributed лицензирования

Установить ecss-elm-adapter

на ecss1

```
sudo apt install ecss-elm-adapter
```

В процессе установки будут заданы вопросы конфигурации , параметры Elm сервиса будут сохранены в конфигурационном файле **/etc/ecss/ecss-elm-adapter/config.exs**.

вопросы	ответы для ecss1	ответы для ecss2	пример
ELM сервис адрес:	Укажите IP адрес или имя сервера в сети для примера: 192.168.83.222 или https://elm.eltex.ru	Укажите IP адрес или имя сервера в сети для примера: 192.168.83.222 или https://elm.eltex.ru	
ELM HTTPS порт:	8099	8099	
Product ID:	для примера: ECSS2000000	для примера: ECSS2000001	
License key:	для примера: ssw987654	для примера: ssw987654	
Log level:	возможные варианты: debug error **info** - по умолчанию	возможные варианты: debug error **info** - по умолчанию	

После установки проверить конфигурацию можно в файле (информация в файлах различается для хостов).

Файл конфигурации находится /etc/ecss/ecss-elm-adapter/config.env.

Пример :

/etc/ecss/ecss-elm-adapter/config.env

```
ELM_HOST=192.168.83.222
ELM_PORT=8099
ECSS_PRODUCT_ID=ECSS2000000
ECSS_LICENSE_KEY=ssw987654
LOG_LEVEL=info
LICENCE_FAIL_TIMER_SECONDS=10
LICENCE_PULL_TIMER_SECONDS=360
```

Чтобы узлы системы смогли встать в работу, надо сконфигурировать систему указав имена хостов на которых развёрнуты ecss-сервисы , для примера ecss1/ecss2.

выполняется в CoCon

```
/system/clusters/set [ecss1, ecss2]
```

выполнить перезагрузку сервисов ECSS

на обоих хостах

```
sudo systemctl restart ecss-*
```

Активировать distributed режим командой:

выполняется в CoCon

```
/cluster/storage/ds1/licence/distributed/apply
```

Пример, список лицензий:

выполняется в CoCon

```
/cluster/storage/ds1/licence/list-licence
┌──┬────────────────────┬─────────────┬──────┬──────────────────┬──────────────────────┬──────────
──────┐
|Id|Creation date(UTC) |   SSW ID    |Active|  Description     |Expiration date(UTC)|
Time left |
├──┼────────────────────┼─────────────┼──────┼──────────────────┼──────────────────────┼──────────
──────┤
|1 |26.12.2025 16:52:27|ECSS 0123456|*     |ECSS ELM License|02.12.2026 07:40:59 |315d
4h 50m|
|0 |                    |ECSS DEFAULT|      |Default licence |                      |
|
└──┴────────────────────┴─────────────┴──────┴──────────────────┴──────────────────────┴──────────
──────┘
```

абонентские лимиты:

выполняется в CoCon

```
/cluster/storage/ds1/licence/subscribers-limit
Licence alias limit: 100000
Licence virtual alias limit: 100000
Active, declared aliases: 22
Active, declared virtual: 0
--------------------------------
Avaliable aliases: 99978
Avaliable virtual aliases: 100000
```

детальная информация

выполняется в CoCon

```
/cluster/storage/ds1/licence/current-limits
Cluster storage
 name                                                ds1
 nodes                                               ds1@ecss2,ds1@ecss1
Cluster core
 name                                                core1
 nodes                                               core1@ecss2,core1@ecss1
Cluster mediator
 name                                                md1
 nodes                                               md1@ecss2,md1@ecss1
Cluster adapter
 name                                                sip1
 type                                                sip
 nodes                                               sip1@ecss2,sip1@ecss1
Cluster adapter
 name                                                megaco1
 type                                                megaco
 nodes                                               megaco1@ecss2,megaco1@ecss1
Cluster adapter
 name                                                sigtran1
 type                                                sigtran
 nodes                                               sigtran1@ecss2,sigtran1@ecss1
Cluster adapter
 name                                                sorm1
 type                                                sorm
 nodes                                               sorm1@ecss2,sorm1@ecss1
Cluster virtual
 name                                                '.system_bridge'
Cluster virtual
 name                                                '.virtual'
TSMN system
 concurrent calls (active)                           10000
 concurrent calls (backup)                           10000
 backup mode                                         none
Teleconference
```

```
 channels                                              50
 max members of one teleconference                     30
 active count                                          5
Supplementary Services licence package(s)
 name                                                  'ECSS-ADV'
 limit                                                 10000
 description                                           "Additional services of 4 and 5
levels"
 SS list                                               [1,2,3,4,5]

 name                                                  'ECSS-BAS'
 limit                                                 10000
 description                                           "Basic services of 1 and 2
levels"
 SS list                                               [1,2]

 name                                                  'ECSS-BAS+'
 limit                                                 10000
 description                                           "Expansion of basic services of
3 level"
 SS list                                               [1,2,3]

SORM
 enabled                                               true
 channels on SORM mediator                             2
SIGTRAN
 SEP                                                   true
 STP                                                   true
Meet Me
 total count                                           5
 members of the one Meet Me conference                 50
IVR
 customization enabled                                 true
 Automatic Speech Recognition (ASR) subsystem enabled  true
GEO backup
 enabled                                               true
Elph
 total count of Elph group                             50
 total count of members of the one Elph group          100
Add-on conferences
 total count                                           50
 members of the one add-on conference                  100
Chat rooms
 total count                                           5
 members of the one chatroom                           50
Call center
 operator's capability for look at calls in queue      true
 operator's capability for use Intervension SS         true
 active agents                                         500
 active supervisors                                    10
Channels on dialer outgoing calls                      100
Total count of simultaneous records voice calls        20000
Support antifraud system                               custom
Sorm extractor                                         mfi
CDR generation mode(s)                                 basic, sorm3
```

```
Virtual subscribers limit                              10000
Subscribers limit                                      10000
System alive time without LPM token (min)              1440
Maximum number of simultaneous calls                   10000
Maximum call duration (in seconds)                     3600
Support for reservation of call-processes              true
```

так же можно проверить статус подключения ECSS к ELM серверу следующей командой, при этом если повторить команду , то запрос будет выполнен со второго хоста кластера.

выполняется в CoCon

```
/cluster/storage/ds1/licence/distributed/show-status
Elm is up (Last checked: "[26.12.2025] 17:14:48")
[exec at: 26.12.2025 17:14:51, exec time: 53ms, nodes: ds1@ecss1]
/cluster/storage/ds1/licence/distributed/show-status
Elm is up (Last checked: "[26.12.2025] 17:14:48")
[exec at: 26.12.2025 17:14:55, exec time: 47ms, nodes: ds1@ecss2]
```

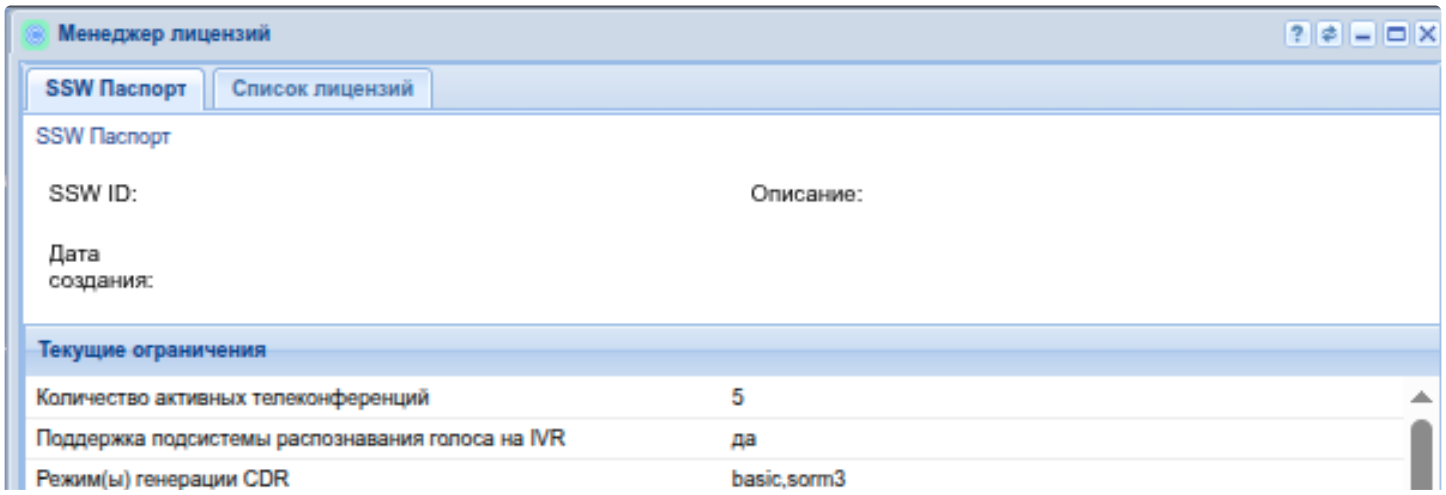
при использовании **distributed** типа лицензирования , следующие команды cocon не будут выполнятся

выполняется в CoCon

```
/cluster/storage/ds1/licence/show-passport
No passport

cluster/storage/ds1/licence/show-licence 1
Invalid command's result: {result,false,{error,parse_failed}}
```

проверить лицензии можно так же через веб интерфейс:



Количество активных агентов call-центра	500
Количество активных супервизоров call-центра	10
Возможность оператора call-центра просматривать вызовы в оче...	да
Возможность оператора call-центра использовать услугу вмешате...	да
Количество подключений до COPM-посредника	2
Количество каналов исходящих вызовов системы автообзвона	100
Узлы кластера bus1 (bus)	bus1@ecss1
Макс. попыток вызова для ядра кластера core1	5
Макс. время разговора для ядра кластера core1, сек	60
Узлы кластера core1 (core)	core1@ecss1
Резервирование кластера core1 (core)	нет
Узлы кластера ds1 (storage)	ds1@ecss1
Узлы кластера md1 (mediator)	md1@ecss1
Узлы кластера megaco1 (adapter)	megaco1@ecss1
Тип кластера megaco1 (adapter)	MEGACO
Узлы кластера sip1 (adapter)	sip1@ecss1
Тип кластера sip1 (adapter)	SIP
Количество активных каналов для вызова на УОВЭОС	10 КБ
Количество резервируемых каналов для вызова на УОВЭОС	10 КБ
GEO резервирование включено	да
Редактор IVR-ограничений включен	да
Максимальное количество участников в телеконференции	30
SEP включен	да
STP включен	да
COPM включен	да
COPM экстрактор	mfi
Количество активных абонентов	10 КБ
Лицензионный пакет(ы) дополнительных услуг	'ECSS-BAS+' (Expansion of basic services of 3 level): 10000 'ECSS-BAS' (Basic services of 1 and 2 levels): 10000 'ECSS-ADV' (Additional services of 4 and 5 levels): 10000
Поддержка антифрод системы	custom
Время работы системы без LPM-токена (мин)	1440
Общее количество участников телеконференций	50
Режим работы 'УОВЭОС резервного транка'	Нет
Общее число конференций с последовательным сбором участни...	50
Общее число чат-комнат	5
Общее число участников одной Meet Me конференции	50
Общее число участников одной конференции с последовательны...	100
Общее число участников одной чат-комнаты	50

Добавить паспорт | Сохранить в файл..

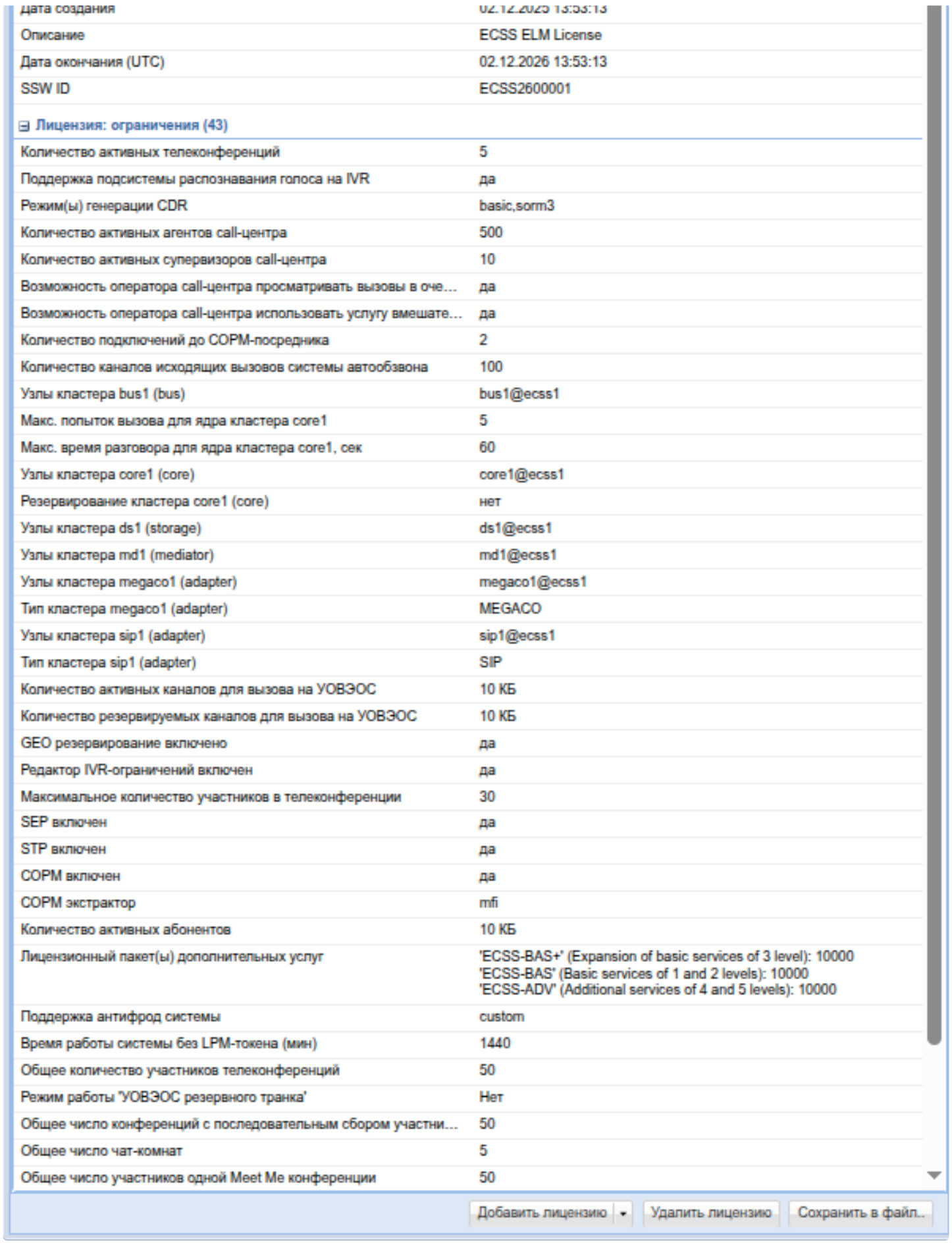
Менеджер лицензий

SSW Паспорт | Список лицензий

Лицензия: 1 ECSS ELM License

Лицензия: информация (5)

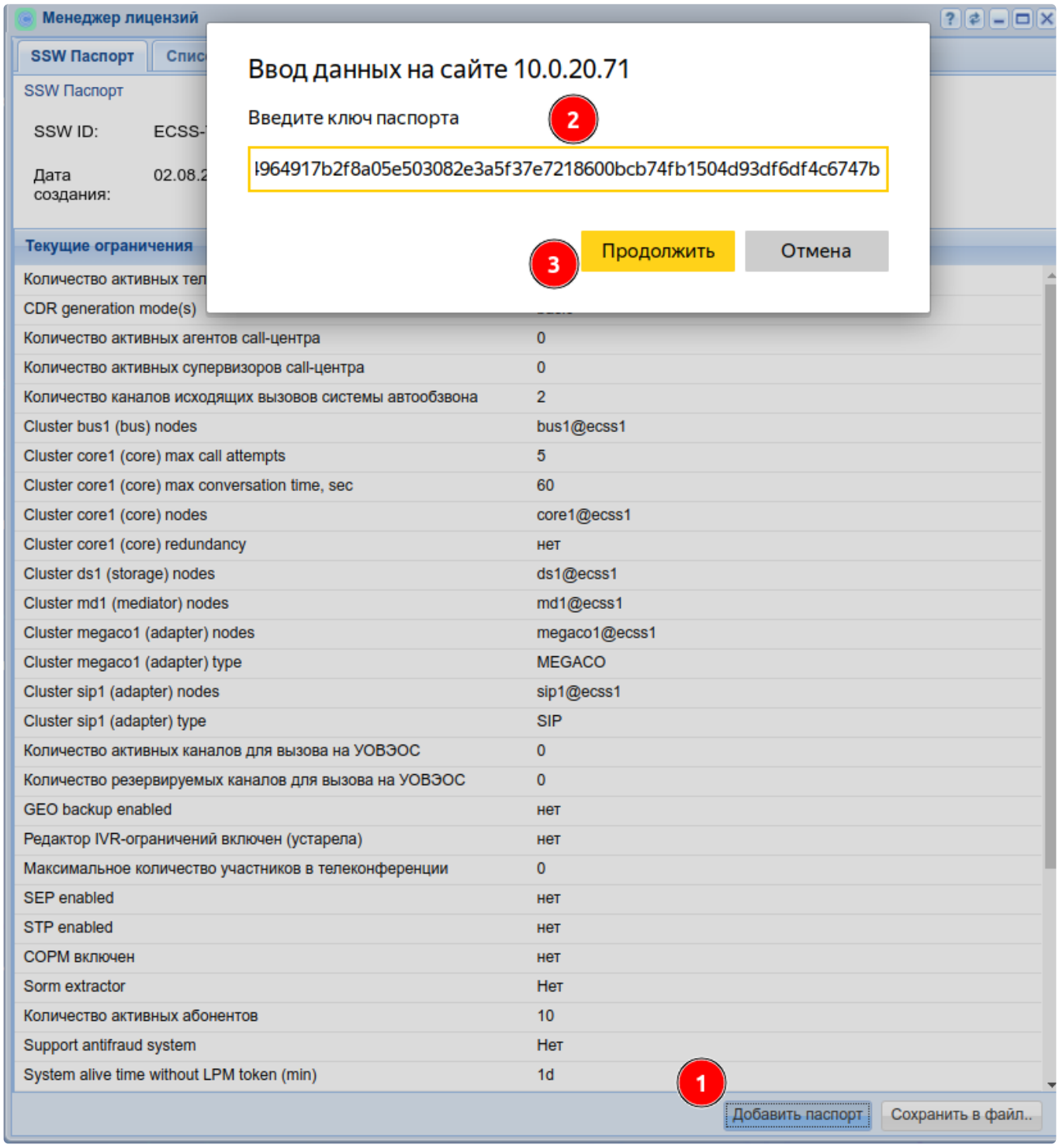
Активная лицензия	да



Дата создания	02.12.2025 13:53:13
Описание	ECSS ELM License
Дата окончания (UTC)	02.12.2026 13:53:13
SSW ID	ECSS2600001
Лицензия: ограничения (43)	
Количество активных телеконференций	5
Поддержка подсистемы распознавания голоса на IVR	да
Режим(ы) генерации CDR	basic,sorm3
Количество активных агентов call-центра	500
Количество активных супервизоров call-центра	10
Возможность оператора call-центра просматривать вызовы в оче...	да
Возможность оператора call-центра использовать услугу вмешате...	да
Количество подключений до COPM-посредника	2
Количество каналов исходящих вызовов системы автообзвона	100
Узлы кластера bus1 (bus)	bus1@ecss1
Макс. попыток вызова для ядра кластера core1	5
Макс. время разговора для ядра кластера core1, сек	60
Узлы кластера core1 (core)	core1@ecss1
Резервирование кластера core1 (core)	нет
Узлы кластера ds1 (storage)	ds1@ecss1
Узлы кластера md1 (mediator)	md1@ecss1
Узлы кластера megaco1 (adapter)	megaco1@ecss1
Тип кластера megaco1 (adapter)	MEGACO
Узлы кластера sip1 (adapter)	sip1@ecss1
Тип кластера sip1 (adapter)	SIP
Количество активных каналов для вызова на УОВЭОС	10 КБ
Количество резервируемых каналов для вызова на УОВЭОС	10 КБ
GEO резервирование включено	да
Редактор IVR-ограничений включен	да
Максимальное количество участников в телеконференции	30
SEP включен	да
STP включен	да
COPM включен	да
COPM экстрактор	mfi
Количество активных абонентов	10 КБ
Лицензионный пакет(ы) дополнительных услуг	'ECSS-BAS+' (Expansion of basic services of 3 level): 10000 'ECSS-BAS' (Basic services of 1 and 2 levels): 10000 'ECSS-ADV' (Additional services of 4 and 5 levels): 10000
Поддержка антифрод системы	custom
Время работы системы без LPM-токена (мин)	1440
Общее количество участников телеконференций	50
Режим работы 'УОВЭОС резервного транка'	Нет
Общее число конференций с последовательным сбором участни...	50
Общее число чат-комнат	5
Общее число участников одной Meet Me конференции	50

Возврат системы на работу с типом лицензирования "**default**" через веб:

1. Нажать кнопку "Добавить паспорт"
2. Ввести ключ паспорта
3. Нажать продолжить



20 Установка Postgres BDR v.3.17

20.1

- Установка
  - Установка вне кластера
    - Установка пакета ecss-postgres-bdr-ssw
    - Проверка
  - Установка в кластере
    - Установка пакета ecss-postgres-bdr-ssw
    - Проверить корректность настройки репликации
- Удаление Postgres BDR
  - Восстановления кластера

20.2 Установка

на всех хостах - Добавление репозитория:

```
sudo sh -c "echo 'deb [arch=amd64] http://apt.ngn.eltex.loc/jammy/3.17 stable main extras external' > /etc/apt/sources.list.d/eltex-ecss10.list"
```

на всех хостах - Импортирование ключа доступа:

```
sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 33CB2B750F8BB6A5
```

Примечание

Перед установкой `ecss-postgres-bdr-ssw` в системе, в `apt sources,` должен быть добавлен `docker` `для этого выполнить следующие команды:`

на всех хостах

```
sudo install -m 0755 -d /etc/apt/keyrings
```

на всех хостах

```
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/
docker.asc
```

на всех хостах

```
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

на всех хостах

```
echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]
https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

на всех хостах - Выполнить обновление:

```
sudo apt update && sudo apt upgrade
```

20.2.1 Установка вне кластера

Установка пакета ecss-postgres-bdr-ssw

ecss1

```
sudo apt install ecss-postgres-bdr-ssw
```

При конфигурации пакета **ecss-postgres-bdr-ssw** будут заданы вопросы:

Вопросы ecss-postgres-bdr-ssw	Ответы для одиночного сервера
Текущий IP-адрес сетевого интерфейса.	**Выбрать** из списка IP-адрес для работы с Postgres

Вопросы ecss-postgres-bdr-ssw	Ответы для одиночного сервера
Выберите <Да>, если пакет устанавливается на узле кластера. Установка в кластере?	No (необходимо выбрать)

Проверка

Проверить порт на котором работает Postgres (должен быть 5439->5432)

```
sudo docker ps
CONTAINER ID   IMAGE                          COMMAND                  CREATED
STATUS         PORTS                                        NAMES
4887950e2633   resistanseltd/postgres-bdr   "/docker-entrypoint.…"   4 minutes ago
Up 4 minutes   0.0.0.0:5439->5432/tcp, [::]:5439->5432/tcp   ecss-postgres-bdr-ssw-
postgres-container-1
```

20.2.2 Установка в кластере

Установка пакета ecss-postgres-bdr-ssw

Выполняется последовательно, первым устанавливается на node работающий в режиме "Maser"

ecss1

```
sudo apt install -y ecss-postgres-bdr-ssw
```

При конфигурации пакета **ecss-postgres-bdr-ssw** будут заданы вопросы:

Вопросы ecss-postgres-bdr-ssw	Ответы для ecss1 (Master)
Текущий IP-адрес сетевого интерфейса.	**Выбрать** из списка IP-адрес для работы с Postgres
Установка в кластере?	Yes (значение по умолчанию)
Является ли этот узел мастером?	**Yes** (необходимо выбрать)

ecss2

```
sudo apt install -y ecss-postgres-bdr-ssw
```

При конфигурации пакета **ecss-postgres-bdr-ssw** будут заданы вопросы:

Вопросы ecss-postgres-bdr-ssw	Ответы для ecss2 (Slave)
Текущий IP-адрес сетевого интерфейса.	**Выбрать** из списка IP-адрес для работы с Postgres
Установка в кластере?	Yes (значение по умолчанию)
Является ли этот узел мастером?	No (значение по умолчанию)
IP-адрес, используемый для подключения к кластеру:	**Указать** IP-адрес Master (ecss1)

При установке на более чем 2 ноды

на дополнительные хосты

```
sudo apt install -y ecss-postgres-bdr-ssw
```

При конфигурации пакета **ecss-postgres-bdr-ssw** будут заданы вопросы:

Вопросы ecss-postgres-bdr-ssw	Ответы для ecssX (Slave)
Текущий IP-адрес сетевого интерфейса.	**Выбрать** из списка IP-адрес для работы с Postgres
Установка в кластере?	Yes (значение **по умолчанию**)
Является ли этот узел мастером?	No (значение **по умолчанию**)
IP-адрес, используемый для подключения к кластеру:	**Указать** IP-адрес Master (ecss1)

Проверить корректность настройки репликации

Во время установки в директорию `/srv/ecss/ecss-postgres-bdr-ssw` устанавливаются скрипт **`postgresbdr_ssw.sh`**

Для проверки репликации выполнить следующие команды:

Для проверки репликации выполнить команду на обоих хостах:

```
cd /srv/ecss/ecss-postgres-bdr-ssw
```

Для проверки репликации выполнить команду на обоих хостах:

```
sudo ./postgresbdr_ssw.sh check
```

Вывод на хостах должен быль одинаковый

При кластере из 3 нод в выводе будет 3 строки

ecss1

```
sudo ./postgresbdr_ssw.sh check
Enter master ip please
Like 10.150.150.15
10.0.10.51
Enter slave ip please
10.0.10.52
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
таблица нод на слейве по ecss_storekeeper_db
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+-----------------------------------------------------------------------------
----------
 7569499877813944343 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.52 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 rows)

таблица коннекций на слейве по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+-----------------------------------------------------------------------------
----------
 7569499877813944343 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.52 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 rows)
```

ecss2

```
sudo ./postgresbdr_ssw.sh check
Enter master ip please
Like 10.150.150.15
10.0.10.51
Enter slave ip please
10.0.10.52
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
таблица нод на слейве по ecss_storekeeper_db
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+----------------------------------------------------------------------------
----------
 7569499877813944343 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.52 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 rows)

таблица коннекций на слейве по ecss_storekeeper_db
     conn_sysid      |                                       conn_dsn

---------------------
+----------------------------------------------------------------------------
----------
 7569499877813944343 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.52 port=5439 dbname=ecss_storekeeper_db
user=postgres password=postgres1
(2 rows)
```

пример с ошибками

нет доступа до slave

```
sudo ./postgresbdr_ssw.sh check
[sudo] password for abf:
Sorry, try again.
[sudo] password for abf:
Enter master ip please
Like 10.150.150.15
10.0.10.51
Enter slave ip please
10.0.10.52
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q – выход из просмотра таблицы (если необходимо)
таблица нод на мастере по ecss_storekeeper_db
     node_sysid      |                                        node_local_dsn
                                    | node_init_from_dsn
----------------------
+-------------------------------------------------------------------------
--------------+--------------------
 7569871659329171478 | host=10.0.10.51 port=5439
dbname=ecss_storekeeper_db user=postgres password=postgres1 |
(1 row)
таблица нод на слейве по ecss_storekeeper_db
psql: error: connection to server at "10.0.10.52", port 5439 failed:
Connection refused
    Is the server running on that host and accepting TCP/IP connections?
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid      |                                          conn_dsn
                                 |
----------------------
+-------------------------------------------------------------------------
--------------
```

7569871659329171478 | host=10.0.10.51 port=5439 dbname=ecss_storekeeper_db user=postgres password=postgres1
(1 row)

таблица коннекций на слейве по ecss_storekeeper_db
psql: error: connection to server at "10.0.10.52", port 5439 failed: Connection refused
    Is the server running on that host and accepting TCP/IP connections?

ошибка несоответствия коннекций

sudo ./postgresbdr_ssw.sh check
Enter master ip please
Like 10.150.150.15
10.0.10.51
Enter slave ip please
10.0.10.52
Press enter with empty password
Просто щелкайте Enter не вводя пароль
q - выход из просмотра таблицы (если необходимо)
таблица нод **на мастере** по ecss_storekeeper_db
     node_sysid | node_local_dsn
                                  | node_init_from_dsn

+---
--------------+--------------------
 7569871659329171478 | host=10.0.10.51 port=5439
dbname=ecss_storekeeper_db user=postgres password=postgres1 |
(1 row)

таблица нод **на слейве** по ecss_storekeeper_db
таблица коннекций на мастер по ecss_storekeeper_db
     conn_sysid | conn_dsn
                                                  |

+---

```
 7569871659329171478 | host=10.0.10.51 port=5439
dbname=ecss_storekeeper_db user=postgres password=postgres1
 (1 row)

 таблица коннекций на слейве по ecss_storekeeper_db
       conn_sysid        |                                              conn_dsn

 ---------------------
+-------------------------------------------------------------------------
--------------
 7569499877813944343 | host=10.0.10.51 port=5439
dbname=ecss_storekeeper_db user=postgres password=postgres1
 7569514001040228374 | host=10.0.10.52 port=5439
dbname=ecss_storekeeper_db user=postgres password=postgres1
 7569871659329171478 | host=10.0.10.51 port=5439
dbname=ecss_storekeeper_db user=postgres password=postgres1
 (3 rows)
```

20.3 Удаление Postgres BDR

Удаление Postgres BDR

удаление пакета ecss-postgres-bdr-ssw

```
sudo apt remove -y ecss-postgres-bdr-ssw
```

При удаления, **чтобы не потерять данные**, остается volume контейнера (находится по пути /srv/ecss/ecss-postgres-bdr-ssw/) имя папки дополняется_дата_время, .

```
 /srv/ecss/ecss-postgres-bdr-ssw$ ll
 total 99692
 drwxr-xr-x 5 root root      4096 ноя  7 13:48 ./
 drwxr-xr-x 3 root root      4096 ноя  7 12:43 ../
 -rwx------ 1 root root      6529 окт 23 14:48
cluster_resolver5439.sh*
```

```
-rw-r--r-- 1 root root       360 ноя  1 11:45 docker-
compose.container.yml
-rw-r--r-- 1 root root 102029198 окт 23 14:48 ecss_postgres_bdr.tar
-rwx------ 1 root root     16657 окт 23 14:48 postgresbdr_ssw.sh*
drwxr-xr-x 3 root root      4096 ноя  7 13:48 volumes/
drwxr-xr-x 3 root root      4096 ноя  7 12:44
volumes_2025_11_07_13_04/
```

Удаление Postgres BDR с очисткой конфигурации:

```
sudo apt purge -y ecss-postgres-bdr-ssw
```

```
/srv/ecss/ecss-postgres-bdr-ssw$ ll
total 99692
drwxr-xr-x 5 root root      4096 ноя  7 13:48 ./
drwxr-xr-x 3 root root      4096 ноя  7 12:43 ../
drwxr-xr-x 3 root root      4096 ноя  7 13:48 volumes_2025_11_07_13_48/
drwxr-xr-x 3 root root      4096 ноя  7 12:44 volumes_2025_11_07_13_04/
```

```
sudo apt autoremove -y
```

20.3.1 Восстановления кластера

Postgres работает в режиме **Multi-master Replication**:

- При которой несколько узлов кластера действуют как мастера.
- Обеспечивать высокую доступность - выход одного из узлов не влияет на доступность базы данных.

BDR (Bi-Directional Replication) основан на функции **логического декодирования** в PostgreSQL, которая извлекает все постоянные изменения из журнала WAL (Write Ahead Log)

и переводит их в форму, специфичную для приложения (поток строк или команды SQL).

ТЕХНИЧЕСКАЯ ПОДДЕРЖКА

Для получения технической консультации по вопросам эксплуатации оборудования ООО «Предприятие «ЭЛТЕКС» вы можете обратиться в Сервисный центр компании:

Форма обратной связи на сайте: https://eltex.ru/support

Servicedesk: https://servicedesk.eltex-co.ru

На официальном сайте компании вы можете найти техническую документацию и программное обеспечение для продукции ООО «Предприятие «ЭЛТЕКС», обратиться к базе знаний или оставить интерактивную заявку:

Официальный сайт компании: https://eltex.ru

База знаний: https://docs.eltex-co.ru/display/EKB/Eltex+Knowledge+Base

Центр загрузок: https://eltex.ru/support/download