ESR series service routers

ESR-10, ESR-12V, ESR-12VF, ESR-14VF, ESR-20, ESR-21, ESR-100, ESR-200, ESR-1000, ESR-1200, ESR-1500, ESR-3100, ESR-1700

CLI command reference guide (29.06.2021)

Firmware version 1.13.0

Contents

1 Introduction 3
2 List of changes 5
3 Command line use specification 22
4 Command system structure 26
5 User interface commands 49
6 Firmware and configuration management 70
7 DNS configuration 98
8 mDNS-reflector configuration 101
9 General-system parameters configuration 105
10 System timer management 118
11 AAA configuration 135
12 Interface monitoring and configuration 188
13 Tunnels configuration and monitoring 301
14 L2 functions management 355
15 Address table operation 400
16 VRF Configuration 414
17 IP addressing configuration 418
18 IPv6 addressing configuration 424
19 Profiles management 437
20 NAT management 455
21 IPsec VPN configuration 480
22 VPN management. Remote access settings 528
23 Routing 560
24 MPLS configuration 760
25 Redundancy 795
26 Security 867
27 QoS management 1012
28 Monitoring and management 1049
29 DHCP Configuration 1149
30 SLA configuration 1199
31 Subscriber control configuration (BRAS) 1232
32 Tunnel Wi-Fi controller configuration 1265
33 VoIP configuration 1279
34 TECHNICAL SUPPORT 1348

1 Introduction

- Abstract
- Target Audience
- Notes and warnings
- Abbreviations used

Abstract

The given guidance describes CLI commands for ESR series service router (hereinafter referred to as router or device) administrator.
Command Line Interface (CLI) allows to perform the device management and monitor its operation and status. You will require the PC application supporting Telnet or SSH protocol operation or direct connection via the console port (e.g. HyperTerminal).

Target Audience

CLI command reference guide is dedicated to the technical staff that performs the ESR series routers configuration and monitoring using the CLI interface. Qualified technical personnel should be familiar with the operation basics of TCP/IP protocol stacks and Ethernet networks design concepts.

Notes and warnings

⚠ Notes contain important information, tips or recommendations on device operation and setup.

❗ Warnings are used to inform the user about harmful situations for the device and the user alike, which could cause malfunction or data loss.

Abbreviations used

- AAA – Authentication, Authorization, Accounting
- ARP – Address Resolution Protocol
- BGP – Border Gateway Protocol
- BRAS – Broadband Remote Access Server
- DHCP – Dynamic Host Configuration Protocol
- DNS – Domain Name System
- DDoS – Distributed Denial of Service
- DoS – Denial of Service
- FIFO – First In, First Out
- FTP – File Transfer Protocol
- FXO – Foreign eXchange Office
- FXS – Foreign eXchange Subscriber
- GRE – Generic Routing Encapsulation
- GRED – Gentle Random Early Detaction
- HTTP – HyperText Transfer Protocol
- HTTPS – HyperText Transfer Protocol Secure
- ICMP – Internet Control Message Protocol
- IKE – Internet Key Exchange
- IDS – Intrusion Detection System
- IP – Internet Protocol
- IP4IP4 – IP in IP

- IPS – Intrusion Prevention System
- IPsec – IP Security
- L2TP – Layer 2 Tunneling Protocol
- L2TPv3 – Layer 2 Tunneling Protocol version 3
- LACP – Link Aggregation Control Protocol
- LAG – Link Aggregation Group
- LDAP – Lightweight Directory Access Protocol
- LLDP – Link Layer Discovery Protocol
- MAC – Media Access Control
- MTU – Maximum Transmission Unit
- mDNS – Multicast Domain Name System
- NAT – Network Address Translation
- NAS – Network Access Server
- NTP – Network Time Protocol
- OSPF – Open Shortest Path First
- PPP – Point-to-Point Protocol
- PPTP – Point-to-Point Tunneling Protocol
- PPPOE – Point-to-point protocol over Ethernet
- PSK – Pre-Shared Key
- QoS – Quality of Service
- RADIUS – Remote Authentication Dial In User Service
- RED – Random early detection
- RIP – Routing Informational Protocol
- SFTP – Secure Shell File Transfer Protocol
- SIP – Session Initiation Protocol
- SLA — Service Level Agreement
- SNMP – Simple Network Management Protocol
- SCP – Secure Copy Protocol
- SP – Strict Priority
- SSH – Secure Shell
- STP – Spanning Tree Protocol
- TACACS – Terminal Access Controller Access Control System
- TFTP – Trivial File Transfer Protocol
- URL – Uniform Resource Locator
- VLAN – Virtual Local Area Network
- VPN –Virtual Private Network
- VRF – Virtual Routing/Forwarding
- VRRP – Virtual Router Redundancy Protocol
- VTI – Virtual Tunnel Interface
- WAN – Wide Area Network
- WINS – Windows Internet Name Service
- WRR – Weighted Round Robin
- XAUTH – eXtended Authentication

2 List of changes

Document version	Issue date	Revisions
Version 1.22	29.06.2021	Changes in sections: - Command system structure - User interface commands - Firmware and configuration management - General-system parameters configuration - Interface monitoring and configuration - L2 functions management - Address table operation - Profiles management - IPsec VPN configuration - Routing -> General routing settings - Routing -> BFD configuration - Security -> Firewall management - Security -> Management of logging and protection against network attacks - Security -> Intrusion Prevention System (IPS/IDS) configuration - QoS management - Monitoring and management -> SYSLOG management - Monitoring and management -> Mirroring configuration - Subscriber control configuration (BRAS) - Tunnel Wi-Fi controller configuration Sections added: - Security -> "Antispam" system configuration

Document version	Issue date	Revisions
Version 1.21	29.10.2020	Changes in sections: - Command system structure - Firmware and configuration management - General-system parameters configuration - System timer management - AAA configuration - Tunnels configuration and monitoring - L2 functions management - Address table operation - Filtering management - Management of logging and protection against network attacks - IPsec VPN configuration - VPN management. Remote access settings - Routing -> BGP configuration - Routing -> IS-IS protocol configuration - MPLS configuration - Redundancy - QoS management - Monitoring and management - DHCP Configuration - SLA configuration - Subscriber control configuration (BRAS) - Intrusion Prevention System (IPS/IDS) configuration

Document version	Issue date	Revisions
Version 1.20	05.06.2020	Changes in sections: - Command system structure - Firmware and configuration management - Tunnels configuration and monitoring - L2 functions management - Firewall management - Management of logging and protection against network attacks - NAT management - IPsec VPN configuration - Routing - MPLS configuration - Redundancy - QoS management - Monitoring and management - Tunnel Wi-Fi controller configuration - TECHNICAL SUPPORT

Document version	Issue date	Revisions
Version 1.19	04.02.2020	Sections added: - mDNS-reflector configuration - MPLS configuration Changes in sections: - Command system structure - User interface commands - AAA configuration - Interface monitoring and configuration - Tunnels configuration and monitoring - L2 functions management - VRF Configuration - Profiles management - Firewall management - IPsec VPN configuration - VPN management. Remote access settings - Routing - QoS management - sFlow management - Monitoring and management - Subscriber control configuration (BRAS) - VoIP configuration
Version 1.18	08.11.2019	Synchronization with firmware version 1.8.2

Document version	Issue date	Revisions
Version 1.17	02.09.2019	Sections added: - 36 Intrusion Prevention System (IPS/IDS) configuration Changes in sections: - 1.4 Abbreviations used - 3.2 Router configuration - 5 Firmware and configuration management - 9 AAA configuration - 10 Interface monitoring and configuration - 11 Tunnels configuration and monitoring - 12 L2 functions management - 13 Address table operation - 14 VRF configuration - 17 Profiles management - 19 Firewall management - 24 VPN management Remote access settings - 20 Filtering management - 25 Routing - 27 QoS management - 30 Monitoring and control - 35 VOIP configuration

Document version	Issue date	Revisions
Version 1.16	15.08.2019	Changes in sections: - 1.4 Abbreviations used - 3.2 Router configuration - 4 User interface commands - 5 Firmware and configuration management - 9 AAA configuration - 10 Interface monitoring and configuration - 11 Tunnels configuration and monitoring - 23 IPsec VPN configuration - 24 VPN management Remote access settings - 25 Routing - 27 QoS management - 30 Monitoring and control - 32 SLA configuration - 33 Subscriber control configuration (BRAS) - 34 Tunnel Wi-Fi controller configuration

Document version	Issue date	Revisions
Version 1.15	05.08.2019	Added ESR-20, ESR-21, ESR-1500, ESR-1510 devices Sections added: - 10.5 Serial interfaces - 20 Filtering management - 32 SLA configuration Changes in sections: - 3 Command system structure - 4 User interface commands - 5 Firmware and configuration management - 7 General-system parameters configuration - 8 System timer management - 9 AAA configuration - 10 Interface monitoring and configuration - 11 Tunnels configuration and monitoring - 12 L2 functions management - 13 Address table operation - 14 VRF configuration - 15 IP addressing configuration - 16 IPv6 addressing configuration - 17 Profiles management - 18 Access control list (ACL) management - 19 Firewall management - 21 Management of logging and protection against network attacks - 22 NAT management - 23 IPsec VPN configuration

Document version	Issue date	Revisions
		- 24 VPN management Remote access settings - 25 Routing - 26 Redundancy - 27 QoS management - 28 Netflow management - 29 Sflow management - 30 Monitoring and control - 31 DHCP configuration
Version 1.14	31.01.2019	Changes in sections: - 11 Tunnels configuration and monitoring
Version 1.12	11.11.2018	Synchronization with firmware version 1.4.1
Version 1.11	03.05.2018	Synchronization with firmware version 1.4.0
Version 1.10	03.05.2017	Changes in sections: - 3.2 Router configuration - 4 User interface commands - 6 General-system parameters configuration - 7 System timer management - 8 AAA configuration - 9.1 Interface management - 11 Tunnel management - 13 Address table management - 19 Firewall management - 21 VPN management IPSec settings - 23.8 BGP configuration - 24.5 MultiWAN configuration - 25 QoS management - 31 Subscriber control configuration (BRAS)

Document version	Issue date	Revisions
Version 1.9	16.12.2016	Sections added: - 23.5 Event tracking object configuration - 23.6 BFD configuration - 23.8 BGP configuration - 31 Subscriber control configuration (BRAS) Changes in sections: - 3.2 Router configuration - 3.3 Types and naming procedure of router interfaces - 3.4 Types and naming procedure of router tunnels - 4 User interface commands - 6 General-system parameters configuration - 8 AAA configuration - 9 Interface monitoring and configuration - 11 Tunnel management - 26 Netflow management - 32 Tunnel Wi-Fi controller configuration
Version 1.8	22.07.2016	Changes in sections: - 7 System timer management - 8 AAA configuration - 9.1 Ethernet interfaces - 11 Tunnel management - 24 VRRP management - 20 NAT management - 14 VRF configuration - 12.1 L2 routing management - 24.3.2 Firewall redundancy configuration

Document version	Issue date	Revisions
Version 1.7	28.01.2015	Sections added: - 22.2 OPENVPN - 30 wiSLA configuration (service quality monitoring system) Changes in sections: - 5.8 Firmware management - 5.14 Firmware management - 6 General-system parameters configuration - 8 AAA configuration - 9.1 Ethernet interfaces - 11.13 Tunnel management - 18.12 ACL management - 22.1 L2TP/PPTP - 22.3 General commands for remote access configuration - 23.7 BGP configuration - 20.8 NAT management - 24.1 VRRP management - 11 Tunnel management - 14.4 VRF configuration - 20.24 NAT management - 24.4 MultiWAN - 26 QoS management - 28.1 SNMP configuration - 23.1 General routing settings - 23.5 IPv4/IPv6 static routes configuration - 23.8 OSFP configuration - 23.9 OSPFv3 configuration - 31 Tunnel Wi-Fi controller configuration

Document version	Issue date	Revisions
		- 25 QoS management
Version 1.6	17.08.2015	Sections added: - 26 Netflow management - 27 Sflow management Changes in sections: - 2 Command line use specification - 3.2 Router configuration - 5 Firmware management - 7 System timer management - 8 AAA configuration - 9.1 Ethernet interfaces - 12.1 L2 routing management - 12.3 VLAN monitoring and configuration - 13 Address table operation - 19 Firewall management - 20 NAT management - 21.2 VPN management IPSec settings - 22 VPN management Remote access settings - 1 General routing settings - 23.2 General commands for routes advertising and reception - 23.3 Policy-based routing (PBR) - 23.7 BGP configuration - 23.8 RIP configuration - 23.9 OSPF configuration - 28.1 SNMP configuration - 28.3 SSH, Telnet access configuration - 29 DHCP configuration

Document version	Issue date	Revisions
Version 1.5	22.06.2015	Sections added: - 9 AAA configuration - 11 IPv6 DHCP client management - 13 IPv6 DHCP relay management - 15 IPv6 DHCP server management - 16.2 TDM(E1) - 17.2 MLPPP - 21 VRF configuration - 23 IPv6 addressing configuration - 25 ACL configuration - 35 Policy-based routing (PBR) - 38 Static IPv6 routes configuration - 42 OSPFv3 configuration - 44 Redundancy configuration Changes in sections: - 4 User interface commands - 5 Firmware management - 14 DHCP server monitoring and configuration - 16.1 Ethernet interfaces - 20 Address table operation - 24 IP addresses and ports profiles management - 26 Firewall management - 34 General commands for routes advertising and reception - 37 Static routes configuration - 39 BGP configuration - 40 RIP configuration - 41 OSPF configuration

Document version	Issue date	Revisions
		- 47 QoS configuration - 50 SNMP configuration

Document version	Issue date	Revisions
Version 1.4	11.03.2015	Sections added: - 11 DHCP agent management - 27 General routing settings - 29 Key ring configuration - 34 VRRP management - 35 DualHoming management - 36 MultiWAN configuration - 37 QoS management - 38 Mirroring configuration - 39 Wi-Fi controller configuration - 40 SNMP configuration Changes in sections: - 1.4 Abbreviations used - 3.2 Router configuration - 3.3 Types and naming procedure of router interfaces - 3.4 Types and naming procedure of router tunnels - 5 Firmware and configuration management - 6 General-system parameters configuration - 7 Access configuration - 12 DHCP server monitoring and configuration - 13 Interface monitoring and configuration - 14 Link Aggregation Group (LAG) management - 20 Firewall management - 21 NAT management - 22 VPN management IKE configuration - 23 VPN management IPsec settings - 25 Tunnel management

Document version	Issue date	Revisions
		- 26 Spanning Tree management - 30 Static routes configuration - 31 BGP configuration - 32 RIP configuration - 33 OSPF configuration - 41 Syslog management

Document version	Issue date	Revisions
Version 1.3	25.11.2014	Sections added: - 7 Access configuration - 13 Link Aggregation Group (LAG) management - 14 VLAN monitoring and configuration - 15 Bridge management - 16 Address table operation - 24 Tunnel management - 25 STP, RSTP, MSTP management - 26.2 General commands for routes advertising - 26.3 BGP configuration - 26.4 RIP configuration - 26.5 OSFP configuration - 27 SYSLOG management Changes in sections: - 3.2 Router configuration - 3.3 Types and naming procedure of router interfaces - 4 User interface commands - 5 Firmware and configuration management - 6 General-system parameters configuration - 12 Interface monitoring and configuration - 18 IP addresses and ports profiles management - 19 Firewall management - 20 NAT management - 22 VPN management IPsec settings - 23 VPN management Remote access settings

Document version	Issue date	Revisions
Version 1.2	26.06.2014	Synchronization with firmware version 1.0.2. Sections added: - 5 General-system parameters configuration - 9 System timer management - 18 VPN management Remote access settings Changes in sections: - 3.2 Router configuration - 4 User interface commands - 6 Firmware and configuration management - 7 Static routes configuration - 10 DHCP client management - 12 Interface monitoring and configuration - 14 Firewall management - 16 VPN management IKE configuration - 17 VPN management IPSec settings
Version 1.1	29.04.2014	Synchronization with firmware version 1.0.1. Section 'Device debugging' has been removed to a separate document.
Version 1.0	20.03.2014	First issue
Firmware version	**1.13.0**	

3 Command line use specification

To simplify the use of the command line, the interface supports automatic command completion. This function is activated when the command is incomplete and the <Tab> character is entered.

Another function that helps to use the command line – context help. At any stage of entering a command, you can get a prompt about the following command elements by entering <?> character.

To simplify the commands, the whole command system has a hierarchical structure. There are special branch commands for transition between levels of the hierarchy. This allows to use brief commands on each level. To designate a current level where a user is located, the system prompt string changes dynamically.

Example:

```
esr> enable Enables the 15 privilege level
```

```
esr# configure Switch to the device configuration mode
```

```
esr(config)#
esr(config)# exit return to the level above
esr#
```

For the ease of command line use, shortcut keys listed in the Table 1 are supported.

Table 1 – Description of CLI shortcut keys

Shortcut key	Description
Ctrl+D	In a nested command mode – exit to the previous command mode (exit command), in a root command mode – exit from CLI (logout command)
Ctrl+Z	Exit to a root command mode (top command)
Ctrl+A	Transition to the beginning of line
Ctrl+E	Transition to the end of line
Ctrl+U	Removal of characters to the left of a cursor
Ctrl+K	Removal of characters to the right of a cursor
Ctrl+C	Line clearing, command execution interruption
Ctrl+W	Removal of a word to the left of a cursor
Ctrl+B	Transition of a cursor one position backwards

Shortcut key	Description
Ctrl+F	Transition of a cursor one position ahead
Ctrl+L	Screen clearing

For ease of reading, page-by-page output of a wealth of information is added.

For example:

```
esr# show running-config
syslog max-files 3
syslog file-size 512
syslog file esr info
syslog console info
interface gigabitethernet 1/0/1
  ip address 10.100.14.1/24
exit
interface gigabitethernet 1/0/1.101
exit
interface gigabitethernet 1/0/2
  ip address 192.168.1.1/24
  ip address 10.100.100.2/24
exit
interface gigabitethernet 1/0/2.150
  ip address 10.150.150.2/24
exit
interface gigabitethernet 1/0/2.151
  ip address 10.151.151.15/24
exit
interface gigabitethernet 1/0/3
  ip address dhcp enable
exit
interface gigabitethernet 1/0/5.55
More? Enter - next line; Space - next page; Q - quit; R - show the rest.
```

To disable page-by-page output, use the following command:

```
esr# terminal datadump
```

Command line interface enables user authorization and restricts access to commands depending on their privilege level, provided by the administrator.

You can create a required amount of users in the system. The required privilege level is specified individually for each of them.

> In factory configuration, the system includes one user with **admin** name and **password** password.

To provide the command interface security, the commands are distributed among 1, 10 and 15 privilege levels:

- 1 level – only device monitoring is available;
- 10 level – device configuration, excluding users creation, device reboot and firmware upload, is available;
- 15 level – no restrictions.

To obtain 15 privilege level:

```
(esr)> enable
(esr)#
```

To return to the initial privilege level:

```
(esr)# disable
(esr)>
```

The system allows multiple users to connect to the device simultaneously.

You can use filtration to reduce the amount of data displayed as a response to user’s requests and to simplify the search of required information. To filtrate the information, add '|' character to the end of command line and use one of the filtration options:

– begin – display all information after the line containing the given pattern;

– include – display all lines containing the given pattern;

– exclude – display all lines not containing the given pattern.

The search pattern can be specified by a regular expression and can contain:

– List of characters. You can specify the list by placing the characters in square brackets '[]'. The matching will be checked by characters included in the list. If the first character of the list is '^', the matching will be checked by any character not included in the list. Examples:

- [-az] - 'a', 'z' and '-';
- [a-z] - all latin letters from 'a' to 'z'.

– Special characters:

- ^ – beginning of a line;
- $ – end of a line;
- . – any character in a line;
- * – zero or more times.

Displaying the «show running-config syslog» command without parameters:

```
esr# show running-config syslog
syslog max-files 3
syslog file-size 512
syslog file default info
```

Displaying the «show running-config syslog» command with the 'begin' parameter:

```
esr# show running-config syslog | begin file-size
syslog file-size 512
syslog file default info
```

Displaying the «show running-config syslog» command with the 'include' parameter:

```
esr# show running-config syslog | include file-size
syslog file-size 512
```

Displaying the «show running-config syslog» command with the 'exclude' parameter:

```
esr# show running-config syslog | exclude file-size
syslog max-files 3
syslog file default info
```

Examples of regular expressions usage:

```
esr# show interfaces status | include "^te.*"
te1/0/1         Up      Down    1500     a8:f9:4b:aa:05:d9
te1/0/2         Up      Down    1500     a8:f9:4b:aa:05:da
esr# show interfaces status | include "^gi1/0/1[2568]"
gi1/0/12        Up      Down    1500     a8:f9:4b:aa:05:cc
gi1/0/15        Up      Down    1500     a8:f9:4b:aa:05:cf
gi1/0/16        Up      Down    1500     a8:f9:4b:aa:05:d0
gi1/0/18        Up      Down    1500     a8:f9:4b:aa:05:d2
esr# show interfaces status | include "^[^tgI -]"
bridge 1        Up      Up      1500     a8:f9:4b:aa:05:c0
bridge 2        Up      Up      1500     a8:f9:4b:aa:05:c0
```

4 Command system structure

- Global mode
- Router configuration
- Types and naming procedure of router interfaces
- Types and naming procedure of router tunnels

The system of ESR router CLI commands is divided into hierarchic levels (sections).

Global mode

Top level of the command hierarchy is shown in the table 2.

Table 2 – Command modes hierarchy (top level)

Level	Entry command	Prompt line view	Exit command
Root mode **(ROOT)**		esr> esr#	exit end
Configuration mode **(CONFIG)**	configure	esr(config)#	
Device debugging mode **(DEBUG)**	debug	esr(debug)#	

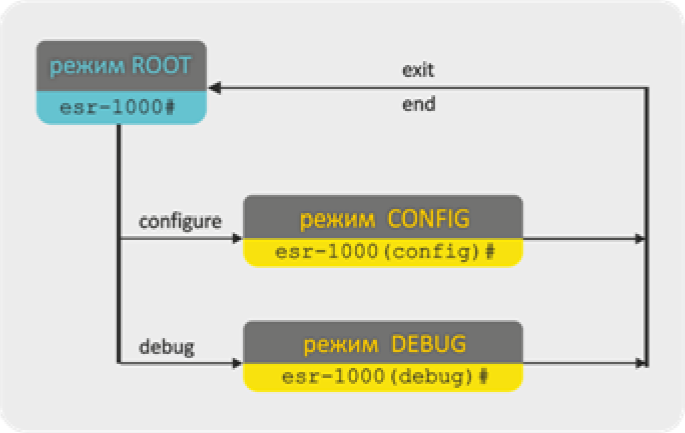

Figure 1 – Top level of command mode hierarchy

In root command mode (ROOT) you can perform the following:

– operation with configuration files:

- application;
- confirmation;
- reset;
- saving;
- undo pending changes;
- revert to the confirmed configuration;

– router reboot;

– monitoring and the current device configuration displaying.

From the root mode (ROOT), proceeding to the following sections is performed:

– device configuration mode (CONFIG);

– device debugging mode (DEBUG).

Router configuration

ESR router configuration is performed in **CONFIG** mode The mode is available from the root mode (ROOT). Switching to the configuration mode is performed only in a privileged one.

To switch from the root mode (ROOT), you should run the following commands:

```
esr> enable
esr# configure
esr(config)#
```

In ESR router configuration mode the following functions are performed:

– system timer management;

– system log management;

– remote access configuration;

– QoS configuration;

– Spanning Tree configuration;

– VLAN configuration;

– static routes configuration;

– routing protocol precedence configuration;

– switching to the function configuration modes, the description is shown in the table 3.

Table 3 – Command modes for router management

Level	Entry command[1]	Prompt line view	Previous level
Global parameters configuration **(CONFIG)**	**configure**	esr(config)#	ROOT
GigabitEthernet interface configuration **(CONFIG-GI)**	**interface gigabitethernet <PORT>**	esr(config-if-gi)#	CONFIG
TengigabitEthernet interface configuration **(CONFIG-TE)**	**interface tengigabitethernet <PORT>**	esr(config-if-te)#	CONFIG
Channel aggregation groups configuration **(CONFIG-PORT-CHANNEL)**	**interface port-channel <CH>**	esr(config-port-channel)#	CONFIG

Level	Entry command[1]	Prompt line view	Previous level
Subinterface configuration **(CONFIG-SUBIF)**	**interface gigabitethernet <PORT>.<VLAN> or** **interface tengigabitethernet <PORT>.<VLAN> or** **interface port-channel <CH>.<VLAN>**	esr(config-subif)#	CONFIG
Subinterface configuration **(CONFIG-QINQ-IF)**	**interface gigabitethernet <PORT>.<VLAN>.<VLAN> or** **interface tengigabitethernet <PORT>.<VLAN>.<VLAN> or** **interface port-channel <CH>.<VLAN>.<VLAN>**	esr(config-qinq-if)#	CONFIG
Serial interface configuration [2] **(CONFIG-SERIAL)**	**interface serial <PORT>**	esr(config-serial)#	CONFIG
Virtual interfaces configuration **(CONFIG-LOOPBACK)**	**interface loopback <PORT>**	esr(config-loopback)#	CONFIG
E1 interface configuration **(CONFIG-E1)**	**interface e1 <PORT>**	esr(config-e1)#	CONFIG
E1 channels aggregation group configuration **(CONFIG-MULTILINK)**	**interface multilink <PORT>**	esr(config-multilink)#	CONFIG
FXO port configuration **(CONFIG-VOICE-PORT-FXO)**	**interface voice-port <NUM>**	esr(config-voice-port-fxo)#	CONFIG
FXS port configuration **(CONFIG-VOICE-PORT-FXS)**	**interface voice-port <NUM>**	esr(config-voice-port-fxs)#	CONFIG
VTI configuration **(CONFIG-VTI)**	**tunnel vti <VTI>**	esr(config-vti)#	CONFIG
Serial interface configuration [2] **(CONFIG-LINE-AUX)**	**line aux <NUM>**	esr(config-line-aux)#	CONFIG

Level	Entry command[1]	Prompt line view	Previous level
L2TPv3 tunnel configuration **(CONFIG-L2TPV3)**	**tunnel l2tpv3 <L2TPV3>**	esr(config-l2tpv3)#	CONFIG
GRE tunnel configuration **(CONFIG-GRE)**	**tunnel gre <GRE>**	esr(config-gre)#	CONFIG
SoftGRE tunnels configuration **(CONFIG-SOFTGRE)**	**tunnel softgre <SOFTGRE>**	esr(config-softgre)#	CONFIG
Logical tunnels configuration **(CONFIG-LT)**	**tunnel lt <LT>**	esr(config-lt)#	CONFIG
Configuration of sub interface on L2 tunnel **(CONFIG-SUBTUNNEL)**	**tunnel softgre <GRE>.<VLAN>**	esr(config-subtunnel)#	CONFIG
IPv4-over-IPv4 tunnel configuration **(CONFIG-IP4IP4)**	**tunnel ip4ip4 <IP4IP4>**	esr(config-ip4ip4)#	CONFIG
Network bridges configuration **(CONFIG-BRIDGE)**	**bridge <BRIDGE>**	esr(config-bridge)#	CONFIG
VLAN Configuration **(CONFIG-VLAN)**	**vlan <VLAN>**	esr(config-vlan)#	CONFIG
Network policy configuration for VLAN specification by LLDPDU messages of a client **(CONFIG-NET-POLICY)**	**network-policy <NAME>**	esr(config-net-policy)	CONFIG
DHCP server's address pool configuration **(CONFIG-DHCP-SERVER)**	**ip dhcp-server pool <NAME>**	esr(config-dhcp-server)#	CONFIG
DHCP option 60 configuration **(CONFIG-DHCP-VENDOR-ID)**	**ip dhcp-server vendor-class-id <NAME>**	esr(config-dhcp-vendor-id)#	CONFIG

Level	Entry command[1]	Prompt line view	Previous level
DHCP server's address pool configuration **(CONFIG-IPV6-DHCP-SERVER)**	**ipv6 dhcp-server pool <NAME>**	esr(config-ipv6-dhcp-server)#	CONFIG
DHCP option 60 configuration **(CONFIG-IPV6-DHCP-VENDOR-ID)**	**ipv6 dhcp-server vendor-class-id <NAME>**	esr(config-dhcp-vendor-id)#	CONFIG
Application profile configuration **(CONFIG-OBJECT-GROUP-APPLICATION)**	**object-group application <NAME>**	esr(config-object-group-application)#	CONFIG
IP address profile configuration **(CONFIG-OBJECT-GROUP-NETWORK)**	**object-group network <NAME>**	esr(config-object-group-network)#	CONFIG
IP address and TCP/UDP port bindings profile configuration **(CONFIG-OBJECT-GROUP-ADDRESS-PORT)**	**object-group address-port <NAME>**	esr(config-object-group-address-port)#	CONFIG
TCP/UDP ports profile configuration **(CONFIG-OBJECT-GROUP-SERVICE)**	**object-group service <NAME>**	esr(config-object-group-service)#	CONFIG
URL profile configuration **(CONFIG-OBJECT-GROUP-URL)**	**object-group url <NAME>**	esr(config-object-group-url)#	CONFIG
Configuration of the content filtering category provider **(CONFIG-OBJECT-GROUP-CONTENT-FILTER)**	**object-group content-filter <NAME>**	esr(config-object-group-content-filter)#	CONFIG
Configuration of the content filtering category profile from the Kaspersky Laboratory vendor **(CONFIG-OBJECT-GROUP-CF-KASPERSKY)**	**vendor <VENDOR>**	esr(config-object-group-cf-kaspersky)#	CONFIG-OBJECT-GROUP-CONTENT-FILTER

Level	Entry command[1]	Prompt line view	Previous level
Configuration of the mail domain profile and mailbox addresses **(CONFIG-OBJECT-GROUP-MAIL)**	**object-group email <NAME>**	esr(config-object-group-email)#	CONFIG
XAUTH profile configuration **(CONFIG-ACCESS-PROFILE)**	**access profile <NAME>**	esr(config-access-profile)#	CONFIG
IP address pool configuration **(CONFIG-POOL)**	**address-assignment pool <NAME>**	esr(config-pool)#	CONFIG
IKE protocol gateway configuration **(CONFIG-IKE-GATEWAY)**	**security ike gateway <NAME>**	esr(config-ike-gw)#	CONFIG
IKE protocol policy configuration **(CONFIG-IKE-POLICY)**	**security ike policy <NAME>**	esr(config-ike-policy)#	CONFIG
IKE protocol profile configuration **(CONFIG-IKE-PROPOSAL)**	**security ike proposal <NAME>**	esr(config-ike-proposal)#	CONFIG
IPsec protocol set policy configuration **(CONFIG-IPSEC-POLICY)**	**security ipsec policy <NAME>**	esr(config-ipsec-policy)#	CONFIG
IPsec protocol set profile configuration **(CONFIG-IPSEC-PROPOSAL)**	**security ipsec proposal <NAME>**	esr(config-ipsec-proposal)#	CONFIG
IPsec-based VPN configuration **(CONFIG-IPSEC-VPN)**	**security ipsec vpn <NAME>**	esr(config-ipsec-vpn)#	CONFIG
Access control list configuration **(CONFIG-ACL)**	**ip access-list extended <NAME>**	esr(config-acl)#	CONFIG
Access control list rule configuration **(CONFIG-ACL-RULE)**	**rule <ORDER>**	esr(config-acl-rule)#	CONFIG-ACL

Level	Entry command[1]	Prompt line view	Previous level
Security zone configuration **(CONFIG-ZONE)**	**security zone <NAME>**	esr(config-zone)#	CONFIG
Rule group configuration for security zone pairs **(CONFIG-ZONE-PAIR)**	**security zone-pair <FROM> <TO>**	esr(config-zone-pair)#	CONFIG
Rule configuration for security zone pairs **(CONFIG-ZONE-PAIR-RULE)**	**rule <ORDER>**	esr(config-zone-rule)#	CONFIG-ZONE-PAIR
Configuration of destination address translation service **(CONFIG-DNAT)**	**nat destination**	esr(config-dnat)#	CONFIG
IP addresses pool and TCP/UDP ports for DNAT configuration **(CONFIG-DNAT-POOL)**	**pool <NAME>**	esr(config-dnat-pool)#	CONFIG-DNAT
DNAT rule group configuration **(CONFIG-DNAT-RULESET)**	**ruleset <NAME>**	esr(config-dnat-ruleset)#	CONFIG-DNAT
DNAT rule configuration **(CONFIG-DNAT-RULE)**	**rule <ORDER>**	esr(config-dnat-rule)#	CONFIG-DNAT-RULESET
Source address translation service configuration **(CONFIG-SNAT)**	**nat source**	esr(config-snat)#	CONFIG
Configuring IP addresses pool and TCP/UDP ports for SNAT **(CONFIG-SNAT-POOL)**	**pool <NAME>**	esr(config-snat-pool)#	CONFIG-SNAT
SNAT rule group configuration **(CONFIG-SNAT-RULESET)**	**ruleset <NAME>**	esr(config-snat-ruleset)#	CONFIG-SNAT

Level	Entry command[1]	Prompt line view	Previous level
SNAT rule configuration **(CONFIG-SNAT-RULE)**	**rule <ORDER>**	esr(config-snat-rule)#	CONFIG-SNAT-RULESET
System users configuration **(CONFIG-USER)**	**username <NAME>**	esr(config-user)#	CONFIG
Local console configuration **(CONFIG-LINE-CONSOLE)**	**line console**	esr(config-line-console)#	CONFIG
Remote console configuration **(CONFIG-LINE-TELNET)**	**line telnet**	esr(config-line-telnet)#	CONFIG
Remote protected console configuration **(CONFIG-LINE-SSH)**	**line ssh**	esr(config-line-ssh)#	CONFIG
TACACS server configuration **(CONFIG-TACACS-SERVER)**	**tacacs-server host <ADDR>**	esr(config-tacacs-server)#	CONFIG
RADIUS server configuration **(CONFIG-RADIUS-SERVER)**	**radius-server host <ADDR>**	esr(config-radius-server)#	CONFIG
RADIUS server profile configuration **(CONFIG-RADIUS-SERVER-PROFILE)**	**aaa radius-profile <NAME>**	esr(config-aaa-radius-profile)#	CONFIG
DAS server configuration **(CONFIG-DAS-SERVER)**	**das-server <NAME>**	esr(config-das-server)#	CONFIG
DAS server profile configuration **(CONFIG-DAS-SERVER-PROFILE)**	**aaa das-profile <NAME>**	esr(config-aaa-das-profile)#	CONFIG
LDAP server configuration **(CONFIG-LDAP-SERVER)**	**ldap-server host <ADDR>**	esr(config-ldap-server)#	CONFIG

Level	Entry command[1]	Prompt line view	Previous level
Configuration of remote access server via PPTP **(CONFIG-PPTP-SERVER)**	**remote-access pptp <NAME>**	esr(config-pptp-server)#	CONFIG
Configuration of remote user access via PPTP **(CONFIG-PPTP-USER)**	**username <NAME>**	esr(config-pptp-user)#	CONFIG-PPTP
Configuration of remote access server via L2TP **(CONFIG-L2TP-SERVER)**	**remote-access l2tp <NAME>**	esr(config-l2tp-server)#	CONFIG
Configuration of remote user access via L2TP **(CONFIG-L2TP-USER)**	**username <NAME>**	esr(config-l2tp-user)#	CONFIG-L2TP-SERVER
Configuration of remote access server via OpenVPN **(CONFIG-OPENVPN-SERVER)**	**remote-access openvpn <NAME>**	esr(config-openvpn-server)#	CONFIG
Configuration of remote user access via L2TP **(CONFIG-OPENVPN-USER)**	**username <NAME>**	esr(config-openvpn-user)#	CONFIG-OPENVPN-SERVER
Configuration of remote access client via PPTP **(CONFIG-PPTP)**	**tunnel pptp <PPTP-ID>**	esr(config-pptp)#	CONFIG
Configuration of remote access client via PPPoE **(CONFIG-PPPOE)**	**tunnel pppoe <PPPOE-ID>**	esr(config-pppoe)#	CONFIG
Configuration of remote connection client via L2TP **(CONFIG-L2TP)**	**tunnel l2tp <L2TP-ID>**	esr(config-l2tp)#	CONFIG
Configuration of remote connection client via OpenVPN **(CONFIG-OPENVPN)**	**tunnel openvpn <OPENVPN-ID>**	esr(config-openvpn)#	CONFIG

Level	Entry command[1]	Prompt line view	Previous level
SNMP user configuration **(CONFIG-SNMP-USER)**	**snmp-server <NAME>**	esr(config-snmp-user)#	CONFIG
NTP server or peer configuration **(CONFIG-NTP)**	**ntp server <ADDR>** **ntp peer <ADDR>**	esr(config-ntp)#	CONFIG
BGP process configuration **(CONFIG-BGP)**	**router bgp <AS>**	esr(config-bgp)#	CONFIG
Configuration of BGP process in vrf **(CONFIG-BGP-VRF)**	**vrf <VRF>**	esr(config-bgp-vrf)#	CONFIG-BGP
Configuration of BGP process IPv4/IPv6 addressing **(CONFIG-BGP-FAMILY)**	**address-family { ipv4 | ipv6 } unicast**	esr(config-bgp-af)#	CONFIG-BGP
Configuration of BGP process IPv4/IPv6 addressing in vrf **(CONFIG-BGP-VRF-FAMILY)**	**address-family { ipv4 | ipv6 } unicast**	esr(config-bgp-vrf-af)#	CONFIG-BGP-VRF
Configuration of BGP neighbor grouping **(CONFIG-BGP-GROUP)**	**peer-group <NAME>**	esr(config-bgp-group)#	CONFIG-BGP
Configuration of BGP neighbor grouping in vrf **(CONFIG-BGP-VRF-GROUP)**	**peer-group <NAME>**	esr(config-bgp-vrf-group)#	CONFIG-BGP-VRF
BGP process neighbor configuration **(CONFIG-BGP-NEIGHBOR)**	**neighbor { <ADDR> | <IPV6-ADDR> }**	esr(config-bgp-neighbor)#	CONFIG-BGP
Configuration of BGP process neighbor in vrf **(CONFIG-BGP-VRF-NEIGHBOR)**	**neighbor { <ADDR> | <IPV6-ADDR> }**	esr(config-bgp-vrf-neighbor)#	CONFIG-BGP-VRF

Level	Entry command[1]	Prompt line view	Previous level
Configuration of BGP neighbor IPv4/ IPv6 addressing **(CONFIG-BGP-NEIGHBOR-FAMILY)**	**address-family { ipv4 | ipv6 | vpnv4 } unicast**	esr(config-bgp-neighbor-af)#	CONFIG-BGP-NEIGHBOR
Configuration of BGP neighbor IPv4/ IPv6 addressing in vrf **(CONFIG-BGP-NEIGHBOR-FAMILY)**	**address-family { ipv4 | ipv6 | vpnv4 } unicast**	esr(config-bgp-neighbor-af-vrf)#	CONFIG-BGP-VRF-NEIGHBOR
Subnet list configuration **(CONFIG-PL)**	**ip prefix-list <NAME>**	esr(config-pl)#	CONFIG
Subnet list configuration **(CONFIG-IPV6-PL)**	**ipv6 prefix-list <NAME>**	esr(config-ipv6-pl)#	CONFIG
Route map configuration **(CONFIG-ROUTE-MAP)**	**route-map <NAME>**	esr(config-route-map)#	CONFIG
Route map rule configuration **(CONFIG-ROUTE-MAP-RULE)**	**rule <ORDER>**	esr(config-route-map-rule)#	CONFIG-ROUTE-MAP
RIP configuration **(CONFIG-RIP)**	**router rip**	esr(config-rip)#	CONFIG
RIPNG configuration **(CONFIG-RIPNG)**	**ipv6 router rip**	esr(config-ripng)#	CONFIG
IS-IS process configuration **(CONFIG-ISIS)**	**router isis <ID>**	esr(config-isis)#	CONFIG
OSPF process configuration **(CONFIG-OSPF)**	**router ospf <ID>**	esr(config-ospf)#	CONFIG
OSPF area configuration **(CONFIG-OSPF-AREA)**	**area <ID>**	esr(config-ospf-area)#	CONFIG-OSPF

Level	**Entry command**[1]	**Prompt line view**	**Previous level**
Virtual OSPF connection configuration **(CONFIG-OSPF-VLINK)**	**virtual-link <ID>**	esr(config-ospf-vlink)#	CONFIG-OSPF-AREA
OSPFv3 process configuration **(CONFIG-OSPFV3)**	**ipv6 router ospf <ID>**	esr(config-ipv6-ospf)#	CONFIG
OSPFv3 area configuration **(CONFIG-OSPFV3- AREA)**	**area <ID>**	esr(config-ipv6-ospf-area)#	CONFIG-OSPFV3
Virtual OSPFv3 connection configuration **(CONFIG-OSPFV3-VLINK)**	**virtual-link <ID>**	esr(config-ipv6-ospf-vlink)#	CONFIG-OSPFV3-AREA
ISIS process configuration **(CONFIG-ISIS)**	**router isis <ID>**	esr(config-isis)#	CONFIG
Key list configuration **(CONFIG-KEYCHAIN)**	**key-chain <KEYCHAIN>**	esr(config-keychain)#	CONFIG
Key configuration **(CONFIG-KEYCHAIN-KEY)**	**key <ID>**	esr(config-keychain-key)#	CONFIG-KEYCHAIN
MSTP parameter configuration **(CONFIG-MST)**	**spanning-tree mst configuration**	esr(config-mst)#	CONFIG
WAN rules configuration **(CONFIG-WAN-RULE)**	**wan load-balance rule <ID>**	esr(config-wan-rule)#	CONFIG
WAN (IPv6) rules configuration **(CONFIG-IPV6-WAN-RULE)**	**Ipv6 wan load-balance rule <ID>**	esr(config-ipv6-wan-rule)#	CONFIG
Target list configuration **(CONFIG-TARGET-LIST)**	**wan load-balance target-list <NAME>**	esr(config-target-list)#	CONFIG

Level	**Entry command**[1]	**Prompt line view**	**Previous level**
Target list (IPv6) configuration **(CONFIG-IPV6-TARGET-LIST)**	**Ipv6 wan load-balance target-list <NAME>**	esr(config-ipv6-target-list)#	CONFIG
Target configuration **(CONFIG-WAN-TARGET)**	**target <ID>**	esr(config-wan-target)#	CONFIG-TARGET-LIST
Target (IPv6) configuration **(CONFIG-IPV6-WAN-TARGET)**	**target <ID>**	esr(config-ipv6-wan-target)#	CONFIG-IPV6-TARGET-LIST
WiFi Controller configuration **(CONFIG-WIRELESS)**	**wireless-controller**	esr(config-wireless)#	CONFIG
VRF Configuration **(CONFIG-VRF)**	**ip vrf <NAME>**	esr(config-vrf)#	CONFIG
QoS policy configuration **(CONFIG-POLICY-MAP)**	**policy-map <NAME>**	esr(config-policy-map)#	CONFIG
QoS class configuration **(CONFIG-CLASS-MAP)**	**class-map <NAME>**	esr(config-class-map)#	CONFIG
Configuration of a class within the QoS policy **(CONFIG-POLICY-MAP-CLASS)**	**class <NAME>**	esr(config-class-policy-map)#	CONFIG
Configuration of a PPP-user to authenticate the remote party **(CONFIG-PPP-USER)**	**ppp chap username <NAME>**	esr(config-ppp-user)#	CONFIG-E1
	ppp chap username <NAME>		CONFIG-MULTILINK
	user <NAME>		CONFIG-CELLULAR-PROFILE
	user <NAME>		CONFIG-ACCESS-PROFILE
	username <NAME>		CONFIG-L2TP-SERVER

Level	Entry command[1]	Prompt line view	Previous level
	username <NAME>		CONFIG-PPTP-SERVER
Redundancy parameters configuration **(CONFIG-ARCHIVE)**	**archive**	esr(config-archive)#	CONFIG
Netflow statistics collection server configuration **(CONFIG-NETFLOW-HOST)**	**netflow collector <ADDR>**	esr(config-netflow-host)#	CONFIG
sFlow statistics collection server configuration **(CONFIG-SFLOW-HOST)**	**sflow collector <ADDR>**	esr(config-sflow-host)#	CONFIG
Configuration of SNMP notification receiving server **(CONFIG-SNMP-HOST)**	**snmp-server host <ADDR>**	esr(config-snmp-host)#	CONFIG
Changing the password after the expiration date **(CHANGE-EXPIRED-PASSWORD)**	**-**	esr(change-expired-password)#	-
Configuration of an http request filtering profile **(CONFIG-PROFILE)**	**ip http profile <NAME>**	esr(config-profile)	CONFIG
MPLS common parameters configuration **(CONFIG-MPLS)**	**mpls**	esr(config-mpls)#	CONFIG
LDP common parameters configuration **(CONFIG-LDP)**	**ldp**	esr(config-ldp)#	CONFIG
LDP IPv4 addressing configuration **(CONFIG-LDP-AF-IPV4)**	**address-family ipv4**	esr(config-ldp-af-ipv4)#	CONFIG-LDP

Level	Entry command[1]	Prompt line view	Previous level
LDP-unicast neighborhood configuration **(CONFIG-LDP-NEIGH)**	**neighbor <ADDR>**	esr(config-ldp-neig)#	CONFIG-LDP
Configuration of LDP IPv4 addressing for a separate ip interface **(CONFIG-LDP-AF-IPV4-IF)**	**interface <IF>**	esr(config-ldp-af-ipv4-if)#	CONFIG-LDP-AF-IPV4
L2-MPLS tunnel creation **(CONFIG-L2VPN)**	**l2vpn**	esr(config-l2vpn)#	CONFIG-MPLS
L2-MPLS tunnel configuration **(CONFIG-L2VPN-EOMPLS)**	**vpls <NAME>** **p2p <NAME>**	esr(config-l2vpn-vpls)# esr(config-l2vpn-p2p)#	CONFIG-L2VPN
PW-class (pseudo-wire pattern) configuration **(CONFIG-L2VPN-PW-CLASS)**	**pw-class <WORD>**	esr(config-l2vpn-pw-class)#	CONFIG-L2VPN
L2-MPLS tunnel pseudo-wire parameters configuration **(CONFIG-L2VPN-PW)**	**pw <PW_ID> <LSR_ID> [<NEIGH_ADDR>]**	esr(config-l2vpn-pw)#	CONFIG-L2VPN-EOMPLS
BGP auto-discovery and singnaling configuration **(CONFIG-AUTODISCOVERY-BGP)**	**autodiscovery bgp**	esr(config-bgp)#	CONFIG-L2VPN-EOMPLS
IPS/IDS policy configuration **(CONFIG-IPS-POLICY)**	**security ips policy <NAME>**	esr(config-ips-policy)#	CONFIG
IPS/IDS rule vendor configuration **(CONFIG-IPS-VENDOR)**	**vendor <VENDOR>**	esr(config-ips-vendor)#	CONFIG-IPS-POLICY
IPS/IDS rule category configuration **(CONFIG-IPS-VENDOR-CATEGORY)**	**category <CATEGORY>**	esr(config-ips-vendor-category)#	CONFIG-IPS-VENDOR

Level	Entry command[1]	Prompt line view	Previous level
IPS/IDS service configuration **(CONFIG-IPS)**	**security ips**	esr(config-ips)#	CONFIG
Configuration of auto-updating source of IPS/IDS rules distributed under a commercial license **(CONFIG-CONTENT-PROVIDER)**	**content-provider**	esr(config-content-provider)#	CONFIG
Configuration of IPS/IDS rules autoupdate from external sources **(CONFIG-IPS-AUTO-UPGRADE)**	**auto-upgrade**	esr(config-ips-auto-upgrade)#	CONFIG-IPS
Configuration of custom IPS/IDS rule update server **(CONFIG-IPS-UPGRADE-USER-SERVER)**	**user-server <WORD>**	esr(config-ips-upgrade-user-server)#	CONFIG-IPS-AUTO-UPGRADE
IPS/IDS user rule category configuration **(CONFIG-IPS-CATEGORY)**	**security ips-category user-defined <CATEGORY_NAME>**	esr(config-ips-category)#	CONFIG
User IPS/IDS rules configuration **(CONFIG-IPS-CATEGORY-RULE)**	**rule <ORDER>**	esr(config-ips-category-rule)#	CONFIG-IPS-CATEGORY
IPS/IDS estended user rule configuration **(CONFIG-IPS-CATEGORY-RULE-ADVANCED)**	**rule-advanced <ORDER>**	esr(config-ips-category-rule-advanced)#	CONFIG-IPS-CATEGORY
Zabbix-agent parameters configuration **(CONFIG-ZABBIX-AGENT)**	**zabbix-agent**	esr(config-zabbix-agent)#	CONFIG
Zabbix-proxy parameters configuration **(CONFIG-ZABBIX-PROXY)**	**zabbix-proxy**	esr(config-zabbix-proxy)#	CONFIG

Level	Entry command[1]	Prompt line view	Previous level
Configuration of the "Antispam" service profile **CONFIG-ANTISPAM-PROFILE**	**security antispam profile <NAME>**	esr(config-antispam-profile)#	CONFIG
Configuration of the rule for the "Antispam" service profile **CONFIG-ANTISPAM-PROFILE-RULE**	**rule <ORDER>**	esr(config-antispam-profile-rule)#	CONFIG
Configuration of the mail domain **CONFIG-MAILSERVER-DOMAIN**	**mailserver domain <NAME>**	esr(config-mailserver-domain)#	CONFIG
Configuration of the mail server **CONFIG-MAILSERVER**	**mailserver**	esr(config-mailserver)#	CONFIG

[1] The detailed description of commands is shown below

[2] Only for ESR-21

Types and naming procedure of router interfaces

Network interfaces of various types and purposes are used for the router operation. The naming system allows you to uniquely address the interfaces by their functional purpose and location in the system. The following table contains the list of interfaces types.

Table 4 – Types and naming procedure of router interfaces

Interface type	Designation
Physical interfaces	Designation of physical interface includes its type and identifier. The identifier of physical interfaces is as follows: **<UNIT>/<SLOT>/<PORT>**, where • **<UNIT>** – number of a device in a device group, • **<SLOT>** – device module number or '0' if the device does not consist of modules, • **<PORT>** – port sequence number.
1Gbps ports	**gigabitethernet <UNIT>/<SLOT>/<PORT>** Designation example: **gigabitethernet 1/0/12** ⚠ It is permitted to use short name, for example, gi1/0/12.
10Gbps ports	**tengigabitethernet <UNIT>/<SLOT>/<PORT>** Designation example: **tengigabitethernet 1/0/2** ⚠ It is permitted to use short name, for example, te1/0/2.
40Gbps ports	**fortygigabitethernet <UNIT>/<SLOT>/<PORT>** Designation example: **forygigabitethernet 1/0/2** ⚠ It is permitted to use short name, for example, fo1/0/2.
Channel aggregation groups	Designation of channel aggregation group includes its type and identifier: **port-channel <CHANNEL_ID>** Designation example: **port-channel 6** ⚠ It is permitted to use short name, for example, po1.

Interface type	Designation
Subinterfaces	Designation of sub-interface is generated from the designation of basic interface and sub-interface identifier (VLAN) separated by a dot. Designation examples: **gigabitethernet 1/0/12.100** **tengigabitethernet 1/0/2.123** **port-channel 1.6** ⚠ Sub-interface identifier may take values of [1..4094].
Q-in-Q interfaces	Designation of Q-in-Q interface is generated from the designation of basic interface, service VLAN identifier and user VLAN identifier separated by a dot. Designation examples: **gigabitethernet 1/0/12.100.10** **tengigabitethernet 1/0/2.45.12** **port-channel 1.6.34** ⚠ Service and user VLAN identifier may take values of [1..4094].
E1 interfaces	Designation of E1 interface includes its type and identifier. E1 interfaces identifier is as follows: **< UNIT>/< SLOT>/< STREAM>**, where • **<UNIT>** – number of a device in a device group [1..1], • **<SLOT>** – number of device E1 module [0..12], • **<STREAM>** – E1 flow sequence number [1..1]. Designation example: **e1 1/0/1**
E1 channels aggregation groups	Designation of E1 channels aggregation group includes its type and interface sequence number: **multilink <CHANNEL_ID>** Designation example: **multilink 3**

Interface type	Designation
Logical interfaces	Designation of logical interface is the interface sequence number: Designation examples: **loopback 4** **bridge 60** **service-port 1**
Serial interfaces	Designation of serial interface includes its type and identifier. E1 interfaces identifier is as follows: **< UNIT>/< SLOT>/< STREAM>**, where • **<UNIT>** – number of a device in a device group [1..1], • **<SLOT>** – device module number or '0' if the device does not consist of modules, • **<PORT>** – port sequence number. Designation example: **serial 1/0/1**
USB modems	Designation of USB modem includes its type and sequence number: **modem <MODEM-NUM>** Designation example: **modem 1**
FXS/FXO ports	Designation of FXS/FXO ports includes its type and sequence number: **interface voice-port <NUM>** Designation example: **voice-port 1**

⚠

1. **Number of interfaces of each type depends on the router model.**
2. **The current firmware does not support for devices stacking. A device number in unit device group can only take the value of 1.**
3. **Some commands support for simultaneous operation with the interface group. To specify the interface group, you may use a comma-separated list or specify a range of identifiers using a hyphen '-'.**

Examples of interface groups specifying:

```
interface gigabitethernet 1/0/1, gigabitethernet 1/0/5
interface tengigabitethernet 1/0/1-2
interface fortygigabitethernet 1/0/1-2
interface gi1/0/1-3,gi1/0/7,te1/0/1,fo1/0/1
```

Types and naming procedure of router tunnels

Network tunnels of various types and purposes are used for the router operation. The naming system allows you to uniquely address the tunnels by their functional purpose. The following table contains the list of tunnels types.

Table 5 – Types and naming procedure of router tunnels

Tunnel type	**Designation**
L2TP tunnel	Designation of L2TP tunnel includes the type and sequence number of a tunnel: **l2tp <L2TP_ID>** Designation example: **l2tp 1**
L2TPv3 tunnel	Designation of L2TPv3 tunnel includes the type and sequence number of a tunnel: **l2tpv3 <L2TPV3_ID>** Designation example: **l2tpv3 1**
GRE tunnel	Designation of GRE tunnel includes the type and sequence number of a tunnel: **gre <GRE_ID>** Designation example: **gre 1**
GRE sub tunnel	Designation of GRE sub tunnel includes the type, sequence number of a tunnel and a sub tunnel VLAN ID: **gre <GRE_ID>.<VLAN_ID>** Designation example: **gre 1.200**
SoftGRE tunnel	Designation of SoftGRE tunnel includes the type and sequence number of a tunnel and, optionally, a virtual interface VLAN ID: **softgre <GRE_ID>[.<VLAN>]** Designation example: **softgre 1, softgre 1.10**
IPv4-over-IPv4 tunnel	Designation of IPv4-over-IPv4 tunnel includes the type and sequence number of a tunnel: **ip4ip4 <IPIP_ID>** Designation example: **ip4ip4 1**
IPsec tunnel	Designation of IPsec tunnel includes the type and sequence number of a tunnel: **vti <VTI_ID>** Designation example: **vti 1**
Logical tunnel (tunnel between VRF)	Designation of logical tunnel includes the type and sequence number of a tunnel: **lt <LT_ID>** Designation example: **lt 1**

Tunnel type	Designation
PPPoE tunnel	Designation of PPPoE tunnel includes the type and sequence number of a tunnel: **pppoe <PPPoE_ID>** Designation example: **pppoe 1**
PPTP tunnel	Designation of PPPTP tunnel includes the type and sequence number of a tunnel: **pptp <PPTP_ID>** Designation example: **pptp 1**

⚠ Number of tunnels of each type depends on the router model and firmware version.

5 User interface commands

- alarm
- clear alarms
- clear ssh host
- configure
- do
- end
- exit
- help
- history size
- logout
- monitor
- ping
- reload system
- show alarms brief
- show alarms brief active
- show alarm settings
- show history
- ssh
- telnet
- terminal datadump
- terminal resize
- traceroute
- uptime
- verify
- verify filesystem

alarm

The command sets alarm threshold values.

Syntax

```
alarm { facility <FACILITY-ALARMS> | memory <MEMORY-ALARMS> | process <PROCESS-ALARMS> }
<VALUE>
```

```
no alarm { facility <FACILITY-ALARMS> | memory <MEMORY-ALARMS> | process <PROCESS-
ALARMS> } <VALUE>
```

Parameters

<FACILITY-ALARMS> – thresholds of alarms connected with the environment (FAN speed, processor and sensors temperature). Takes the following values:

- fan-speed high – fan speed threshold expressed as a percentage of the maximum fan speed (0..100), when crossing which from the lowest values to the highest ones, SNMP trap eltexEnvFanSpeedHigh (alarm) is generated. The value of this threshold should be greater than alarm facility fan-speed low (for esr-100/200/1000/1200/1500/1700/3100);
- fan-speed high – fan speed threshold expressed as a percentage of the maximum fan speed (0..100), when crossing which from the highest values to the lowest ones, SNMP trap eltexEnvFanSpeedHighOk (alarm recovery) is generated. The value of this threshold should be lower than alarm facility fan-speed high (for esr-100/200/1000/1200/1500/1511/1700/3100);
- temperature cpu critical high – processor temperature threshold expressed as degrees Celsius (0..255), when crossing which from the lowest values to the highest ones, SNMP trap eltexEnvTempCritical

(alarm) is generated. The value of this threshold should be the highest of all processor temperature thresholds;

- temperature cpu critical low – processor temperature threshold expressed as degrees Celsius (0..255), when crossing which from the highest values to the lowest ones, SNMP trap eltexEnvTempCriticalOk (alarm recovery) is generated. The value of this threshold should be less than alarm facility temperature cpu critical high but more than alarm facility temperature cpu overheat high;
- temperature <sensor> overheat high – sensor temperature threshold expressed as degrees Celsius (0..255), when crossing which from the lowest values to the highest ones, SNMP trap eltexEnvTempOverheat (alarm) is generated. The value of this threshold should be less than alarm facility temperature <sensor> critical low but more than alarm facility temperature <sensor> overheat low;
- temperature <sensor> overheat low – sensor temperature threshold expressed as degrees Celsius (0..255), when crossing which from the highest values to the lowest ones, SNMP trap eltexEnvTempOverheatOk (alarm recovery) is generated. The value of this threshold should be less than alarm facility temperature <sensor> overheat high;
- temperature <sensor> supercooling high – sensor temperature threshold expressed as degrees Celsius (0..255), when crossing its opposite value from the lowest values to the highest ones, SNMP trap eltexEnvTempSupercoolingOk (alarm recovery) is generated. This threshold is intended to specify negative temperatures, its value should be less than alarm facility temperature <sensor> supercooling low;
- temperature <sensor> supercooling low – sensor temperature threshold expressed as degrees Celsius (0..255), when crossing its opposite value from the highest values to the lowest ones, SNMP trap eltexEnvTempSupercooling (alarm) is generated. This threshold is intended to specify negative temperatures, its value should be more than alarm facility temperature <sensor> supercooling high;
  The 'sensor' field may take the following values per each router model:
  - ESR-10: cpu;
  - ESR-12V: cpu, switch;
  - ESR-12VF/14VF: cpu, switch, sfp;
  - ESR-20/21: cpu;
  - ESR-100/200: cpu, board;
  - ESR-1000/1200/1500/1511/1700: cpu, board, switch, sfp;
  - ESR-3100: cpu, board, sfp, phy.

<MEMORY-ALARMS> – alarm thresholds connected with available NAND and RAM space. Takes the following values:

- free low-watermark flash high – free flash memory threshold expressed as a percentage of the maximum one (0..100), when crossing which from the lowest values to the highest ones, SNMP trap eltexEnvMemoryLowOk (alarm recovery) is generated. The value of this threshold should be the highest of all free flash memory amount thresholds;
- free low-watermark flash low – free flash memory threshold expressed as a percentage of the maximum one (0..100), when crossing which from the highest values to the lowest ones, SNMP trap eltexEnvMemoryLow (alarm) is generated. The value of this threshold should be less than alarm memory free low-watermark flash high;
- free low-watermark ram high – free RAM threshold expressed as a percentage of the maximum one (0..100), when crossing which from the lowest values to the highest ones, SNMP trap eltexEnvMemoryLowOk (alarm recovery) is generated. The value of this threshold should be the highest of all free RAM amount thresholds;
- free low-watermark ram low – free RAM threshold expressed as a percentage of the maximum one (0..100), when crossing which from the highest values to the lowest ones, SNMP trap eltexEnvMemoryLow (alarm) is generated. The value of this threshold should be less than alarm memory free low-watermark ram high;
- free low-watermark flash high – free flash memory threshold expressed as a percentage of the maximum one (0..100), when crossing which from the lowest values to the highest ones, SNMP trap

  eltexEnvMemoryLowOk (alarm recovery) is generated. The value of this threshold should be less than alarm memory free low-watermark flash low but more than alarm memory reserve critical flash low;
- free low-watermark flash high – free flash memory threshold expressed as a percentage of the maximum one (0..100), when crossing which from the lowest values to the highest ones, SNMP trap eltexEnvMemoryCriticalLow (alarm) is generated. The value of this threshold should be less than alarm memory reserve critical flash high;
- reserve critical ram high – free RAM threshold expressed as a percentage of the maximum one (0..100), when crossing which from the lowest values to the highest ones, SNMP trap eltexEnvMemoryCriticalLowOk (alarm recovery) is generated. The value of this threshold should be less than alarm memory free low-watermark ram low but more than alarm memory reserve critical ram low;
- free low-watermark ram high – free RAM threshold expressed as a percentage of the maximum one (0..100), when crossing which from the lowest values to the highest ones, SNMP trap eltexEnvMemoryCriticalLow (alarm) is generated. The value of this threshold should be less than alarm memory reserve critical ram high.

<PROCESS-ALARMS> – alarm threshold connected with the processor utilization. May take the following values:

- cpu threshold falling – CPU utilization threshold expressed as a percentage of the maximum utilization (0..100), when crossing which from the highest values to the lowest ones, SNMP trap eltexEnvCpuLoadHighOk (alarm recovery) is generated. The value of this threshold should be less than alarm process cpu threshold rising;
- cpu threshold rising – CPU utilization threshold expressed as a percentage of the maximum utilization (0..100), when crossing which from the lowest values to the highest ones, SNMP trap eltexEnvCpuLoadHigh (alarm) is generated. The value of this threshold should be more than alarm process cpu threshold falling;

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# alarm facility fan-speed high 80
```

clear alarms

This command is used to clear alarm entries.

Syntax

```
clear alarms { inactive | all }
```

Parameters

inactive – only archive alarm entries are deleted;

all – archive and active alarm entries are deleted.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# clear alarms inactive
```

clear ssh host

The command resets saved SSH key of a remote host.

Syntax

```
clear ssh host { <ADDR> | <IPV6-ADDR> }
```

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

15

Command mode

ROOT

Example:

```
esr# clear ssh host 192.168.1.1
```

configure

This command allows to switch to the global configuration mode.

Syntax

```
configure
configure terminal
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# configure
esr(config)#
```

do

The command allows to execute root mode commands (ROOT) from any other command-line interface mode.

Syntax

`do <command>`

Parameters

<command> – root mode command.

Required privilege level

1

Command mode

All modes except the root one.

Example:

```
esr(config)# do show version
Boot version:
  1.0.7.16 (date 18/11/2015 time 13:40:59)
SW version:
  1.0.7 build 17[d9bdbda] (date 21/11/2015 time 18:06:41)
HW version:
  1v7
```

end

The command allows you to return to the root command mode (ROOT).

Syntax

```
end
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

All modes except the root one.

exit

This command is used to return to a higher level in the hierarchical system of command modes.

When this command is executed in the ROOT mode, the user session with the CLI command line interface of the router is terminated.

Syntax

```
exit
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

All modes.

help

The command displays information on working with the command line.

Syntax

```
help
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

All modes.

history size

The command changes the maximum amount of the latest entered user commands which are kept in the command log of a current session.The use of a negative form (no) of the command sets the default value.

Syntax

```
history size <SIZE>
no history size
```

Parameters

<SIZE> – amount of the latest entered commands, takes values of [10..1000].

Default value

50

Required privilege level

1

Command mode

ROOT

Example:

```
esr# history size 20
```

logout

The command finishes a user session with the CLI interface.

Syntax

```
logout
```

Parameters

The command does not contain parameters

Required privilege level

1

Command mode

ROOT

CHANGE-EXPIRED-PASSWORD

Example:

```
esr# logout
```

monitor

The command enables monitoring of traffic on a network interface in real-time mode packet by packet.

Syntax

```
monitor { <IF> | <TUN> } [ protocol <TYPE> [ source-port <SRC-PORT> ] [ destination-port
<DST-PORT> ] [ port <PORT> ] ] [ source-address {<SRC-ADDR> | <SRC-IPV6-ADDR> } ]
[ destination-address { <DST-ADDR> | <DST-IPV6-ADDR> } ] [address { <ADDR> | <IPV6-
ADDR> } ] [ packets <VALUE> ] [ detailed ]
```

Parameters

<IF> – an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

<SRC-ADDR> – source IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<DST-ADDR> – destination IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<SRC-IPV6-ADDR> – source IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<DST-IPV6-ADDR> – destination IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<ADDR> – sender or receiver IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – sender or receiver IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<SRC-MAC> – ethernet frame source mac address, specified as HH:HH:HH:HH:HH:HH, where every HH part may take values in the hexadecimal format [0..FF];

<DST-MAC> – ethernet frame destination mac address, specified as HH:HH:HH:HH:HH:HH, where every HH part may take values in the hexadecimal format [0..FF];

<SRC-MAC> – ethernet frame source or destination mac address, specified as HH:HH:HH:HH:HH:HH, where every HH part may take values in the hexadecimal format [0..FF];

<TYPE> – protocol type, takes the following values: tcp, udp, icmp, icmp6, igrp, igmp, arp, gre, ipip, esp, ah, eigrp, ospf, pim, vrrp, l2tp, RDP or protocol number [0..255];

<SRC-PORT> – number of source TCP/UDP port, takes values of [1..65535];

<DST-PORT> – number of destination TCP/UDP port, takes values of [1..65535];

<PORT> – number of sender or receiver TCP/UDP port, takes values of [1..65535];

<VALUE> – amount of packets after receiving which the evaluating will be stopped, defined in the range of [1…4294967295];

detailed – information is given in the detailed format.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# monitor gigabitethernet 1/0/5 detailed
23:37:44.324049 d8:50:e6:d2:f0:46 > a8:f9:4b:aa:03:a5, ethertype IPv4 (0x0800), length 98: (tos
0x0, ttl
 64, id 50760, offset 0, flags [DF], proto ICMP (1), length 84)
    10.255.100.1 > 10.255.100.5: ICMP echo request, id 11730, seq 19, length 64
```

ping

The command is used to check the availability of a specified network device.

Syntax

```
ping [ vrf <VRF>] { <ADDR> | ip { <vDDR> | <HOSTNAME> } | ipv6 { <IPV6-ADDR> |
<HOSTNAME> } } [ ttl <TTL> ] [ packets <COUNT> |unlimited ] [ size <SIZE> ] [ timeout
<TIMEOUT> ] [interval <INTERVAL>] [ source { ip { <SRC-ADDR> | <SRC-IPV6-ADDR> } |
interface <IF> | tunnel <TUN> } ] [ data <HEX> ] [ dscp <DSCP> ] [ flood ] [detailed]
[ strategy <STRATEGY> ] [ nodeinfo <INFO> ] [ broadcast ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters;

<ADDR> – device IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – device IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<HOSTNAME> – user DNS name, set by the string of up to 255 characters;

<TTL> – IP packet life time, takes value in the range of [1..255], 64 by default;

<COUNT> – number of packets being transmitted [1..4294967295], default is 5;

<SIZE> – packet size in bytes, takes the value of [1..65468], 56 bytes by default which corresponds to 64 bytes after adding ICMP header;

<TIMEOUT> – response time out, in seconds. The option affects the timeout if there are no any responds, otherwise the utility waits for RTTs. Takes values in the range of [1..60], 1 second by default;

<INTERVAL> – interval between sending icmp packets in milliseconds, takes a value of [200..60000], default is 1000.

<SRC-ADDR> – source IP address, any router IP address may be used as the given address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<SRC-IPV6-ADDR> – source IPv6 address, any router IPv6 address may be used as the given address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<IF> – name of router interface through which the packets will be sent, specified in the form which is described in Section Types and naming order of router interfaces;

<TUN> – name of router tunnel through which the packets will be sent, specified in the form which is described in Section Types and naming order of router tunnels;

<HEX> – pattern of data through which a packet will be filled, defined as a hexadecimal number up to 16 bytes;

<DSCP> – DSCP priority according to RFC 2474, takes the value of [0..63], default value is 0;

flood – when specifying this command, the packets will be sent with the maximum rate, responses from the device are not displayed till the end of the command execution;

broadcast – when specifying this command, it will be permitted to send packets to the broadcast address;

detailed – when specifying this key, complete information about icmp answers and general statistics of the command work will be displayed. Without specifying – only statistics.

<STRATEGY> – packet fragmentation strategy, takes on of the following values:

- allow-fragmentation – allow fragmentation, do not set the DF (don't fragment) flag;
- discovery-pmtu – discover PMTU (Path MTU), fragment locally if a packet size is too big;
- disallow-fragmentation – disallow fragmentation including the local one.

<INFO> – only for IPv6. Sending of ICMPv6 Node Information Queries (RFC4620), instead of Echo Request takes one of the following values:

- name – DNS host name request;
- ipv6 – host IPv6 addresses request;
- ipv6-global – request for host global IPv6 addresses;
- ipv6-sitelocal – request for host site-local IPv6 addresses;
- ipv6-linklocal – request for host link-local IPv6 addresses;
- ipv6-all – request for host unicast IPv6 addresses;
- ipv4 – request for host IP addresses;
- ipv4-all – request for IP addresses from all host network interfaces.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# ping 192.168.100.39 packets 5 size 1400 detailed
PING 192.168.100.39 (192.168.100.39) 1400(1428) bytes of data.
1408 bytes from 192.168.100.39: icmp_req=1 ttl=64 time=0.084 ms
1408 bytes from 192.168.100.39: icmp_req=2 ttl=64 time=0.053 ms
1408 bytes from 192.168.100.39: icmp_req=3 ttl=64 time=0.082 ms
1408 bytes from 192.168.100.39: icmp_req=4 ttl=64 time=0.051 ms
1408 bytes from 192.168.100.39: icmp_req=5 ttl=64 time=0.075 ms
--- 192.168.100.39 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3999ms
rtt min/avg/max/mdev = 0.051/0.069/0.084/0.014 ms
esr# ping ipv6 fc00::1
PING fc00::1(fc00::1) 56 data bytes
64 bytes from fc00::1: icmp_seq=1 ttl=64 time=0.379 ms
64 bytes from fc00::1: icmp_seq=2 ttl=64 time=0.161 ms
--- fc00::1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.161/0.270/0.379/0.109 ms
```

reload system

The command performs the device reboot.

Syntax

```
reload system
```

Parameters

The command does not contain parameters

Required privilege level

15

Command mode

ROOT

Example:

```
esr# reload system
```

show alarms brief

This command displays a brief history of crashes on the router (active and completed).

Syntax

`show alarms brief`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# show alarms brief

   History Alarms
   ~~~~~~~~~~~~~~
Severity   Group              Set time            Clear time          Description
--------   ---------------    -----------------   -----------------
---------------------------
major      environment        03/31/00 17:27:38   03/31/00 17:31:53   Reserve Power Supply Fault
```

show alarms brief active

This command displays a summary of current (active) alarms on the router.

Syntax

`show alarms brief active`

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show alarms brief active
   History Alarms
   ~~~~~~~~~~~~~~
Severity   Group              Set time           Clear time         Description
--------   ---------------    -----------------  -----------------
-----------------------------------
notify     environment        03/31/00 16:47:05  -                  Reserve Power Supply
removed
```

show alarm settings

The command displays the information on alarm threshold settings.

Syntax

`show alarm settings <TYPE>`

Parameters

<TYPE> – alarm type:

- facility – thresholds of alarms connected with the environment (FAN speed, processor and sensors temperature);
- memory – thresholds of alarms connected with the available NAND and RAM space;
- process – thresholds of alarms connected with the processor utilization;

Required privilege level

10

Command mode

ROOT

Example:

```
esr# show alarm facility
    fan-speed:
        high:                    80
        low:                     75
    temperature CPU:
        critical high:           95
        critical low:            92
        overheat high:           80
        overheat low:            78
        supercooling high:       -12
        supercooling low:        -15
    temperature sensor1:
        overheat high:           60
        overheat low:            57
        supercooling high:       -12
        supercooling low:        -15
    temperature sensor2:
        overheat high:           60
        overheat low:            57
        supercooling high:       -12
        supercooling low:        -15
    temperature sensor3:
        overheat high:           60
        overheat low:            57
        supercooling high:       -12
        supercooling low:        -15
```

show history

The command displays information on the command used in a current session or on the number of stored commands.

Syntax

```
show history [size | <NUM> ]
```

Parameters

size – maximum amount of the latest entered user commands which are kept in the command log of a current session.

<NUM> – the number of last entered user commands to be displayed [0..100]

Default value

1. Corresponds to the conclusion of the entire history of the entered commands.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show history
     1  enable
     2  show history
     3  configure
     4  service nat
     5  service nat source
     6  exit
     7  show history
```

ssh

The command is used for connection to a remote host via SSH.

Syntax

```
ssh [ vrf <VRF>] <USERNAME> { <ADDR> | <IPV6-ADDR> | <HOSTNAME> } [ port <PORT> ]
[ version <VERSION> ] [ source { <SRC-ADDR> | <SRC-IPV6-ADDR> } ] [ dscp <DSCP> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters.

<USERNAME> – user name, set by the string of up to 31 characters.

<ADDR> – device IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<IPV6-ADDR> – device IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

<HOSTNAME> – device DNS name, set by the string of up to 255 characters.

<PORT> – number of TCP port listened by SSH server, takes values of [1..65535]. 22 is set by default.

<VERSION> – SSH protocol version, takes values in the range of [1..2]. Version 1 is used by default.

<SRC-ADDR> – source IP address, any router IP address may be used as the given address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<SRC-IPV6-ADDR> – source IPv6 address, any router IPv6 address may be used as the given address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

<DSCP> – DSCP priority according to RFC 2474, takes the value of [0..63], default value is 0.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# ssh tester 10.100.100.1
The authenticity of host '10.100.100.1 (10.100.100.1)' can't be established.
ECDSA key fingerprint is db:e4:0a:93:59:87:7d:9f:90:5c:19:a3:e7:97:ec:d5.
Are you sure you want to continue connecting (yes/no)? yes
%AAA-I-SSH: Warning: Permanently added '10.100.100.1' (ECDSA) to the list of known hosts.
tester@10.100.100.1's password:
Welcome to Ubuntu 14.04.2 LTS (GNU/Linux 3.13.0-51-generic x86_64)
 * Documentation:  https://help.ubuntu.com/
  System information as of Mon May 25 09:25:10 NOVT 2015
Last login: Tue May 12 19:39:11 2015
(tester@kubuntu  ~) $
```

telnet

The command is used for connection to a remote host via Telnet.

Syntax

```
telnet [ vrf <VRF>] { <ADDR> | <IPV6-ADDR> | <HOSTNAME> } [ port <PORT> ] [ source
{ <SRC-ADDR> | <SRC-IPV6-ADDR> } ] [ dscp <DSCP> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters;

<ADDR> – device IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – device IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<HOSTNAME> – user DNS name, set by the string of up to 255 characters;

<PORT> – number of TCP port listened by SSH server, takes values of [1..65535], default value is 23;

<SRC-ADDR> – source IP address, any router IP address may be used as the given address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<SRC-IPV6-ADDR> – source IPv6 address, any router IPv6 address may be used as the given address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<DSCP> – DSCP priority according to RFC 2474, takes the value of [0..63], default value is 0.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# telnet 10.100.100.1
Entering character mode
Escape character is '^]'.
Ubuntu 14.04.2 LTS
kubuntu login: tester
Password:
Last login: Mon May 25 15:23:06 NOVT 2015 from sw31-1.eltex.loc on pts/16
Welcome to Ubuntu 14.04.2 LTS (GNU/Linux 3.13.0-51-generic x86_64)
 * Documentation:  https://help.ubuntu.com/
  System information as of Mon May 25 15:23:01 NOVT 2015
(tester@kubuntu  ~) $
```

terminal datadump

The command is used to disable page-by-page trace displaying for a current session.

The use of the negative command enables page-by-page trace displaying.

Syntax

`[no] terminal datadump`

Parameters

The command does not contain parameters

Required privilege level

1

Command mode

ROOT

Example:

```
esr# terminal datadump
```

terminal resize

The command is used to scale the terminal size to fit the window size when using a console connection.

Syntax

`terminal resize`

Parameters

The command does not contain parameters

Required privilege level

1

Command mode

ROOT

Example:

```
esr# terminal resize
```

traceroute

The command is used to trace a route to a specified network device.

Syntax

```
traceroute [ vrf <VRF> ] { <ADDR> | ip { <ADDR> | <HOSTNAME> } | ipv6 { <IPV6-ADDR> |
<HOSTNAME> } } [ first-ttl <FIRST-TTL> ] [ max-ttl <MAX-TTL> ] [ timeout <TIMEOUT> ]
[ source { ip { <SRC-ADDR> | <SRC-IPV6-ADDR> } | interface <IF> | tunnel <TUN> } ] [ dscp
<DSCP> ] [ protocol { icmp | udp [ <PORT> ] | tcp [ <PORT> ] } ] [ gateway { <GW-ADDR> |
<GW-IPV6-ADDR> } ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters;

<ADDR> – device IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – device IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<HOSTNAME> – user DNS name, set by the string of up to 255 characters;

<FIRST-TTL> – IP packet life time, the value from which a route tracing starts, takes values in the range of [1..255], 1 by default;

<FIRST-TTL> – IP packet life time, the value at which a route tracing stops, takes values in the range of [1..255], 30 by default;

<TIMEOUT> – timeout when waiting for the respond, in seconds. The option affects the timeout if there are no any responds, otherwise the utility waits for RTTs. Takes values in the range of [1..60], 5 seconds by default;

<SRC-ADDR> – source IP address, any router IP address may be used as the given address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<SRC-IPV6-ADDR> – source IPv6 address, any router IPv6 address may be used as the given address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<IF> – name of router interface through which the packets will be sent, specified in the form which is described in Section Types and naming order of router interfaces;

<TUN> – name of router tunnel through which the packets will be sent, specified in the form which is described in Section Types and naming order of router tunnels;

<DSCP> – DSCP priority according to RFC 2474, takes the value of [0..63], default value is 0;

<PORT> – TCP/UDP port number, takes the value of [1..65535], default value is 53 for UDP and 80 for TCP;

<GW-ADDR> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. When specifying the parameter, IP source routing option, signalling to the router through which gateway the packet should be routed in a network, is added to an outgoing packet. For reasons of safety, the routing via this option is disabled on the most routers;

<GWIPV6-ADDR> – gateway IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF]. When specifying the parameter, IP source routing option, signalling to the router through which gateway the packet should be routed in a network, is added to an outgoing packet. For reasons of safety, the routing via this option is disabled on the most routers.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# traceroute 192.168.27.128
traceroute to 192.168.27.128 (192.168.27.128), 30 hops max, 60 byte packets
1 192.168.16.1 (192.168.16.1)  1.240 ms  1.546 ms  1.883 ms
2 192.168.27.128 (192.168.27.128)  0.451 ms  0.437 ms  0.411 ms
```

uptime

The command displays the device operation time.

Syntax

`uptime`

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# uptime
System uptime:         26 minutes and 35 seconds
```

verify

This command calculates the hash sum for a single file using the specified hashing algorithm.

Syntax

```
verify <ALGORITHM> <FILE>
```

Parameters

<ALGORITHM> – hashing algorithm, takes values of: [md5, sha2-256, sha2-512].

<FILE> – path and file name to verify, can take the following values:

- usb://usb_name:/PATH
- mmc://mmc_name:/PATH (except esr-10)
- system:candidate-config
- system:running-config
- system:factory-config
- system:default-config
- system:firmware-image-1
- system:firmware-image-2
- system:boot-1
- system:boot-2
- flash:critlog/FILE
- flash:syslog/FILE
- flash:data/FILE
- tmpsys:syslog/FILE

Required privilege level

15

Command mode

ROOT

Example:

```
esr# verify md5 system:firmware-image-1
system:firmware-image-1 16ef38a292e96ce972e910da6db2d1f4
```

verify filesystem

This command starts the process of calculating the hash sums for all the system files of the router and comparing them with the reference values. As a result, information on the compliance of the calculated hash amounts with the reference is displayed.

Syntax

```
verify filesystem [detailed]
```

Parameters

detailed – key, that responsible for the detailed display of information about the verification of each file.

Required privilege level

15

Command mode

ROOT

Example:

```
esr# verify filesystem
Total:          2949
Success:        2949
Changed:        0
Deleted:        0


Filesystem verification success.
```

6 Firmware and configuration management

- archive
- auto
- boot system
- by-commit
- commit
- confirm
- copy
- count-backup
- delete
- dir
- merge
- path
- restore
- rollback
- save
- show bootvar
- show boot-licence
- show candidate-config
- show configuration changes
- show crypto certificates
- show licence
- show running-config
- show storage-devices
- show version
- time-period
- type

archive

The command performs the switch to configuration redundancy parameters mode.

The use of a negative form (no) of the command sets the default value for configuration redundancy parameters.

Syntax

```
[no] archive
```

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# archive
esr(config-archive)#
```

auto

This command enables the creation of a backup configuration file on a remote server and/or locally after a specified period of time (see section time-period).

The use of a negative form (no) of the command disables sending mode in a specified period of time.

Syntax

```
[no] auto
```

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

CONFIG-ARCHIVE

Example:

```
esr(config-archive)# auto
```

boot system

The command is used to select an active software image loaded to the device.

Syntax

```
boot system <IMAGE>
```

Parameters

<IMAGE> – name of the software image which will be loaded to the device:

- image-1 – next device boot will be performed from the first software image;
- image-2 – next device boot will be performed from the second software image.

Required privilege level

15

Command mode

ROOT

Example:

```
esr# boot system image-2
```

by-commit

The command enables the mode of sending the configuration file to the redundancy server in a specified period of time.

The use of a negative form (no) of the command disables the sending mode after the successful application of configuration.

Syntax

`[no] by-commit`

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

CONFIG-ARCHIVE

Example:

```
esr(config-archive)# by-commit
```

commit

The command allows to apply (enable) the configuration changes. RUNNING configuration is replaced by CANDIDATE configuration. To enable the changes applied, you need to confirm the operation by ‘confirm’ command during the time period not exceeding the acknowledgement timer lifetime (600 seconds by default, can be changed by the system config-confirm timeout command).

Syntax

`commit`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

CHANGE-EXPIRED-PASSWORD

Example:

```
esr# commit
```

Configuration changes are applied.

confirm

The command is intended to confirm the configuration appliance. If during a specified time period (600 seconds by default, can be changed by the system config-confirm timeout command) after applying the configuration you do not enter a confirmation by «commit» command, automatic rollback to a previously valid configuration will occur. Automatic rollback system prevents loss of connection with the device.

Syntax

```
confirm
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

CHANGE-EXPIRED-PASSWORD

Example:

```
esr# confirm
```

Confirmation of configuration changes.

copy

The command is used to copy files among the various sources and receivers.

Syntax

```
copy <SOURCE> <DESTINATION>
```

Parameters

<SOURCE> – source, defined as:

- tftp://<ip>[<port>]:/<path> – address of file on TFTP server, where:
  - <ip> – TFTP server IP address;
  - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
  - <path> – file path on TFTP server.
- tftp://<ipv6>[%<interface>][<port>]:/<path> – address of file on TFTP server, where:
  - <ipv6> – TFTP server IPv6 address;
  - <interface> – outgoing network interface for link-local addresses;
  - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on TFTP server.
- ftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – FTP server IP address;
  - <user> – user name (configure the default user name by the *ip ftp client username* command, described in Section ip ftp client username);
  - <password> – password (configure the default password by the *ip ftp client password* command, described in Section ip ftp client password);
  - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
  - <path> – file path on FTP server.
- ftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – FTP server IPv6 address;
  - <interface> – outgoing network interface for link-local addresses;
  - <user> – user name (configure the default user name by the *ip ftp client username* command, described in Section ip ftp client username);
  - <password> – password (configure the default password by *ip ftp client password* command, described in Section ip ftp client password);
  - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on FTP server.
- sftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – FTP server IP address;
  - <user> – user name (configure the default user name by the *ip sftp client username* command, described in Section ip sftp client username);
  - <password> – password (configure the default password by the *ip sftp client password* command, described in Section ip sftp client password);
  - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
  - <path> – file path on FTP server.
- sftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – FTP server IPv6 address;
  - <interface> – outgoing network interface for link-local addresses;
  - <user> – user name (configure the default user name by the *ip sftp client username* command, described in Section ip sftp client username);

  - <password> – password (configure the default password by the *ip sftp client password* command, described in Section ip sftp client password);
  - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on FTP server.
- scp://[<user>:<password>@]<ip>[<port>]:/<path>
  - <ip> – IP address of server;
  - <user> – user name (configure the default user name by the *ip ssh client username* command, described in Section ip ssh client username);
  - <password> – password (configure the default password by *ip ssh client password* command, described in Section ip ssh client password);
  - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
  - <path> – file path on server.
- scp://[<user>:<password>@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6 address of server;
  - <interface> – outgoing network interface for link-local addresses;
  - <user> – user name (configure the default user name by the *ip ssh client username* command, described in Section ip ssh client username);
  - <password> – password (configure the default password by the *ip ssh client password* command, described in Section ip ssh client password);
  - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on server.
- http://<ip>[<port>]:/<path> – address of file on HTTP server, where:
  - <ip> – IP address of HTTP server;
  - <port> – port on which HTTP server is launched, separated from IP address by '#' or ':' character;
  - <path> – file path on HTTP server.
- http://<ipv6>[%<interface>][<port>]:/<path> – address of file on HTTP server, where:
  - <ipv6> – IPv6 address of HTTP server;
  - <interface> – outgoing network interface for link-local addresses;
  - <port> – port on which HTTP server is launched is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on TFTP server.
- https://[ca/<CA-FILE>][:server-crt/<SC-FILE>][:server-key/<SK-FILE>]@<ip>[<port>]:/<path> – address of file on HTTPS server, where:
  - <CA-FILE> – authentication server certificate file name in the corresponding memory section of the router;
  - <SC-FILE> – file name of the server public certificate in the corresponding section of the router's memory;
  - <SK-FILE> – server private key file name in the corresponding memory section of the router;
  - <ip> – IP address of HTTP server;
  - <port> – port on which HTTPS server is launched, separated from IP address by '#' or ':' character;
  - <path> – file path on HTTP server.
- https://[ca/<CA-FILE>][:server-crt/<SC-FILE>][:server-key/<SK-FILE>]@<ipv6>[%<interface>][<port>]:/<path> – address of file on HTTPS server, where:
  - <CA-FILE> – authentication server certificate file name in the corresponding memory section of the router;
  - <SC-FILE> – file name of the server public certificate in the corresponding section of the router's memory;
  - <SK-FILE> – server private key file name in the corresponding memory section of the router;
  - <ipv6> – IPv6 address of HTTPS server;
  - <interface> – outgoing network interface for link-local addresses;
  - <port> – port on which HTTP server is launched is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');

  - <path> – file path on TFTP server.
- usb://usb_name:/PATH
  - usb_name – name assigned to USB storage. The following parameters can be displayed by "show storage-devices" command (see Section show storage-devices);
  - <PATH> – file path on USB storage.
- mmc://mmc_name:/PATH (except ESR-10/12V/12VF/14VF)
  - mmc_name – name assigned to MMC storage. The following parameters can be displayed by "show storage-devices" command (see Section show storage-devices);
  - <PATH> – file path on MMC storage.
- system:factory-config – factory configuration;
- system:default-config – default configuration (blank);
- system:running-config – running configuration;
- system:candidate-config – configuration which will be applied after 'commit' command execution;
- system:firmware – device firmware. Copying is performed from inactive device software image;
- system:boot-1 – primary device bootloader (SBI, bl1, x-loader).
- system:boot-2 – secondary device bootloader (u-boot, boot).
- flash:critlog/FILE – folder to save kernel messages during the entire operation of the device;
- flash:syslog/FILE – folder to save current session logs, saved after rebooting;
- tmpsys:syslog/FILE – folder to save current session logs, is not saved after rebooting;
- flash:backup/FILE – folder for saving backup copies of current router configurations;
- flash:data/FILE – folder for downloading files from the router.

<DESTINATION> – destination, defined as:

- tftp://<ip>[<port>]:/<path> – address of file on TFTP server, where:
  - <ip> – TFTP server IP address;
  - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
  - <path> – file path on TFTP server.
- tftp://<ipv6>[%<interface>][<port>]:/<path> – address of file on TFTP server, where:
  - <ipv6> – TFTP server IPv6 address;
  - <interface> – outgoing network interface for link-local addresses;
  - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on TFTP server.
- ftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – FTP server IP address;
  - <user> – user name (configure the default user name by the *ip ftp client username* command, described in Section ip ftp client username);
  - <password> – password (configure the default password by *ip ftp client password* command, described in Section ip ftp client password);
  - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
  - <path> – file path on FTP server.
- ftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – FTP server IPv6 address;
  - <interface> – outgoing network interface for link-local addresses;
  - <user> – user name (configure the default user name by the *ip ftp client username* command, described in Section ip ftp client username);
  - <password> – password (configure the default password by the *ip ftp client password* command, described in Section ip ftp client password);
  - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on FTP server.
- sftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – FTP server IP address;

  - <user> – user name (configure the default user name by the *ip sftp client username* command, described in Section ip sftp client username);
  - <password> – password (configure the default password by the *ip sftp client password* command, described in Section ip sftp client password);
  - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
  - <path> – file path on FTP server.
- sftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – FTP server IPv6 address;
  - <interface> – outgoing network interface for link-local addresses;
  - <user> – user name (configure the default user name by the *ip sftp client username* command, described in Section ip sftp client username);
  - <password> – password (configure the default password by *the ip sftp client password* command, described in Section ip sftp client password);
  - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on FTP server.
- scp://[<user>:<password>@]<ip>[<port>]:/<path>
  - <ip> – IP address of server;
  - <user> – user name (configure the default user name by the *ip ssh client username* command, described in Section ip ssh client username);
  - <password> – password (configure the default password by *ip ssh client password* command, described in Section ip ssh client password);
  - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
  - <path> – file path on server.
- scp://[<user>:<password>@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6 address of server;
  - <interface> – outgoing network interface for link-local addresses;
  - <user> – user name (configure the default user name by the *ip ssh client username* command, described in Section ip ssh client username);
  - <password> – password (configure the default password by *ip ssh client password* command, described in Section ip ssh client password);
  - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on server.
- usb://usb_name:/PATH
  - usb_name – name assigned to USB storage. The following parameters can be displayed by "show storage-devices" command (see Section show storage-devices);
  - <PATH> – file path on USB storage.
- mmc://mmc_name:/PATH (except ESR-10/12V/12VF/14VF)
  - mmc_name – name assigned to MMC storage. The following parameters can be displayed by "show storage-devices" command (see Section show storage-devices);
  - <PATH> – file path on MMC storage.
- system:candidate-config – configuration which will be applied after 'commit' command execution;
- system:licence – device licence;
- system:boot-licence – device license used when starting the secondary bootloader;
- system:firmware – device firmware. Copying is always performed to inactive device software image;
- system:boot2 – secondary device bootloader (u-boot, x-loader);
- certificate/ca/ – folder to keep CA's certificate;
- certificate/dh/ – folder to keep Diffie-Hellman key;
- certificate/server_key/ – folder to keep a private key of OPENVPN server;
- certificate/server_crt/ – folder to keep a private key of OPENVPN server;
- certificate:client-key/FILE – folder to keep a private key of OPENVPN client;
- certificate:client-crt/FILE – folder to keep a certificate of OPENVPN client;
- certificate/ta/ – folder to keep HMAC key;
- certificate/crl/ – folder to keep the list of revoked certificates;

- flash:data/FILE – folder for downloading files from the router.

Required privilege level

15

Command mode

ROOT

Example 1

```
esr# copy tftp://10.100.100.1/esr.cfg system:candidate-config
```

Example 2

```
esr# copy tftp://10.100.100.1/crl.pem system:certificate/crl/crl.pem
```

count-backup

This command sets the maximum number of locally saved configuration backups.

The use of a negative form (no) of the command sets the default value.

Syntax

```
count-backup <NUM>
no count-backup
```

Parameters

<NUM> – set the maximum number of locally saved configuration backups. Takes values in the range of [1..100].

Default value

1

Required privilege level

15

Command mode

CONFIG-ARCHIVE

Example:

```
esr(config-archive)# count-backup 20
```

delete

The command is used to remove licences, certificates, keys.

Syntax

`delete <FILE>`

Parameters

<FILE> – file type, may take the following values (when removing from the folder, you need to specify a file name):

- flash:backup/FILE – folder for saving backup copies of current router configurations;
- flash:critlog/FILE – folder to save kernel messages during the entire operation of the device;
- flash:data/FILE – folder for downloading files from the router;
- flash:syslog/FILE – folder to save current session logs, saved after rebooting;
- tmpsys:syslog/FILE – folder to save current session logs, is not saved after rebooting;
- certificate/ca/FILE – folder to keep CA's certificate;
- certificate/dh/FILE – folder to keep Diffie-Hellman key;
- certificate/server_key/FILE – folder to keep a private key of IPsec/OpenVPN server key;
- certificate/server_crt/FILE – folder to keep a public certificate of IPsec/OpenVPN server;
- certificate:client-key/FILE – folder to keep a private key of OPENVPN client;
- certificate:client-crt/FILE – folder to keep a certificate of OPENVPN client;
- certificate/ta/FILE – folder to keep HMAC key;
- certificate/crl/FILE – folder to keep the list of revoked certificates;
- system:licence – license to activate functionality that requires a license;
- system:boot-licence – device license used when starting the secondary bootloader;
- usb://usb_name:/FILE
  - usb_name – name assigned to USB storage. The following parameters can be displayed by "show storage-devices" command (see Section show storage-devices);
  - <FILE> – the path to the file and its name on the USB flash drive.
- mmc://mmc_name:/<FILE> (except ESR-10/12V/12VF/14VF)
  - mmc_name – name assigned to MMC storage. The following parameters can be displayed by "show storage-devices" command (see Section show storage-devices);
  - <FILE> – the path to the file and its name on the MMC flash drive.

Required privilege level

15

Command mode

ROOT

Example:

```
esr# delete system:certificate/dh/dh.key
```

dir

The command displays external storage medium (USB/MMC cards) and local router content.

Syntax

```
dir <PATH>
{ usb://<USB-device-name>/<PATH> | mmc://<MMC-device-name>/<PATH> | flash:backup/FILE |
flash:data/FILE }
```

Parameters

<PATH> – the name of the local partition or the path to the folder on the external drive. Takes the following values:

- flash:critlog – folder to save kernel messages during the entire operation of the device;
- flash:syslog/ – folder to save current session logs, saved after rebooting;
- flash:backup/ – folder for saving backup copies of current router configurations;
- flash:data/ – folder for downloading files from the router;
- tmpsys:syslog/ – folder to save current session logs, is not saved after rebooting;
- certificate/ca/ – folder to keep CA's certificate;
- certificate/dh/ – folder to keep Diffie-Hellman key;
- certificate/ta/ – folder to keep HMAC key
- certificate/crl/ – folder to keep the list of revoked certificates;
- certificate/server_key/ – folder to keep a private key of IPsec/OpenVPN server key;
- certificate/server_crt/ – folder to keep a public certificate of IPsec/OpenVPN server;
- certificate:client-key/ – folder to keep a private key of OPENVPN client;
- certificate:client-crt/ – folder to keep a certificate of OPENVPN client;
- usb://usb_name:/PATH
  - usb_name – name assigned to USB storage. The following parameters can be displayed by "show storage-devices" command (see Section show storage-devices);
  - <PATH> – file path on USB storage.
- mmc://mmc_name:/PATH (except ESR-10/12V/12VF/14VF)
  - mmc_name – name assigned to MMC storage. The following parameters can be displayed by "show storage-devices" command (see Section show storage-devices);
  - <PATH> – file path on MMC storage.

<USB-device-name> – USB storage name assigned by the system. Displayed in 'show storage-devices usb' command output, Section show storage-devices.

<MMC-device-name> – MMC storage name assigned by the system. Displayed in 'show storage-devices usb' command output, Section show storage-devices.

- flash:backup/FILE – file for saving backup copies of current router configurations;
- flash:data/FILE – folder for downloading files from the router;

<PATH> – path to the required folder on USB/MMC storage.

flash:backup/FILE – file for saving backup copies of current router configurations;

flash:data/FILE – folder for downloading files from the router;

Required privilege level

1

Command mode

ROOT

Example:

```
esr# dir mmc://EF28-D074
Name                                       Type         Size
----------------------------------------   ----------   --------   --
esr1000-1.4.0-build21.uboot                File         0.00       B
.Trash-1000                                Directory    0.00       B
```

merge

This command is used to merge a local or downloadable configuration file with candidate-config.

Syntax

```
merge <SOURCE> system:candidate-config
```

Parameters

<SOURCE> – source, defined as:

- tftp://<ip>[<port>]:/<path> – address of file on TFTP server, where:
  - <ip> – TFTP server IP address;
  - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
  - <path> – file path on TFTP server.
- tftp://<ipv6>[%<interface>][<port>]:/<path> – address of file on TFTP server, where:
  - <ipv6> – TFTP server IPv6 address;
  - <interface> – outgoing network interface for link-local addresses;
  - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on TFTP server.
- ftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – FTP server IP address;
  - <user> – user name (configure the default user name by the *ip ftp client username* command, described in Section ip ftp client username);
  - <password> – password (configure the default password by the *ip ftp client password* command, described in Section ip ftp client password);
  - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
  - <path> – file path on FTP server.
- ftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – FTP server IPv6 address;
  - <interface> – outgoing network interface for link-local addresses;
  - <user> – user name (configure the default user name by the *ip ftp client username* command, described in Section ip ftp client username);

  - <password> – password (configure the default password by *ip ftp client password* command, described in Section ip ftp client password);
  - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on FTP server.
- sftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – FTP server IP address;
  - <user> – user name (configure the default user name by the *ip sftp client username* command, described in Section ip sftp client username);
  - <password> – password (configure the default password by the *ip sftp client password* command, described in Section ip sftp client password);
  - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
  - <path> – file path on FTP server.
- sftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – FTP server IPv6 address;
  - <interface> – outgoing network interface for link-local addresses;
  - <user> – user name (configure the default user name by the *ip sftp client username* command, described in Section ip sftp client username);
  - <password> – password (configure the default password by the *ip sftp client password* command, described in Section ip sftp client password);
  - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on FTP server.
- scp://[<user>:<password>@]<ip>[<port>]:/<path>
  - <ip> – IP address of server;
  - <user> – user name (configure the default user name by the *ip ssh client username* command, described in Section ip ssh client username);
  - <password> – password (configure the default password by *ip ssh client password* command, described in Section ip ssh client password);
  - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
  - <path> – file path on server.
- scp://[<user>:<password>@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6 address of server;
  - <interface> – outgoing network interface for link-local addresses;
  - <user> – user name (configure the default user name by the *ip ssh client username* command, described in Section ip ssh client username);
  - <password> – password (configure the default password by the *ip ssh client password* command, described in Section ip ssh client password);
  - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on server.
- http://<ip>[<port>]:/<path> – address of file on HTTP server, where:
  - <ip> – IP address of HTTP server;
  - <port> – port on which HTTP server is launched, separated from IP address by '#' or ':' character;
  - <path> – file path on HTTP server.
- http://<ipv6>[%<interface>][<port>]:/<path> – address of file on HTTP server, where:
  - <ipv6> – IPv6 address of HTTP server;
  - <interface> – outgoing network interface for link-local addresses;
  - <port> – port on which HTTP server is launched is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on TFTP server.
- https://[ca/<CA-FILE>][:server-crt/<SC-FILE>][:server-key/<SK-FILE>]@<ip>[<port>]:/<path> – address of file on HTTPS server, where:

    - <CA-FILE> – authentication server certificate file name in the corresponding memory section of the router;
    - <SC-FILE> – file name of the server public certificate in the corresponding section of the router’s memory;
    - <SK-FILE> – server private key file name in the corresponding memory section of the router;
    - <ip> – IP address of HTTP server;
    - <port> – port on which HTTPS server is launched, separated from IP address by '#' or ':' character;
    - <path> – file path on HTTP server.
- https://[ca/<CA-FILE>][:server-crt/<SC-FILE>][:server-key/<SK-FILE>]@<ipv6>[%<interface>][<port>]:/<path> – address of file on HTTPS server, where:
    - <CA-FILE> – authentication server certificate file name in the corresponding memory section of the router;
    - <SC-FILE> – file name of the server public certificate in the corresponding section of the router’s memory;
    - <SK-FILE> – server private key file name in the corresponding memory section of the router;
    - <ipv6> – IPv6 address of HTTPS server;
    - <interface> – outgoing network interface for link-local addresses;
    - <port> – port on which HTTP server is launched is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
    - <path> – file path on TFTP server.
- flash:data/FILE – folder for downloading files from the router.

Required privilege level

15

Command mode

ROOT

Example

```
esr# merge tftp://10.100.100.1/esr.cfg system:candidate-config
```

path

The command defines the protocol, server address as well as the location and name prefix of a file on the server. When performing a redundancy, the current time and date in the format YYYYMMDD_HHMMSS is added to the file name prefix.

The use of a negative form (no) of the command removes a specified value.

Syntax

`path <PATH>`

`no path`

Parameters

<PATH> – the format of the path to the folder on the remote server by tftp/ftp/sftp/scp in one of the following formats:

- tftp://<ip>[<port>]:/<path> – address of file on TFTP server, where:
    - <ip> – TFTP server IP address;
    - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
    - <path> – file path on TFTP server.
- tftp://<ipv6>[%<interface>][<port>]:/<path> – address of file on TFTP server, where:
    - <ipv6> – TFTP server IPv6 address;
    - <interface> – outgoing network interface for link-local addresses;
    - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
    - <path> – file path on TFTP server.
- ftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
    - <ip> – FTP server IP address;
    - <user> – user name (configure the default user name by the *ip ftp client username* command, described in Section ip ftp client username);
    - <password> – password (configure the default password by the *ip ftp client password* command, described in Section ip ftp client password);
    - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
    - <path> – file path on FTP server.
- ftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
    - <ipv6> – FTP server IPv6 address;
    - <interface> – outgoing network interface for link-local addresses;
    - <user> – user name (configure the default user name by the *ip ftp client username* command, described in Section ip ftp client username);
    - <password> – password (configure the default password by *ip ftp client password* command, described in Section ip ftp client password);
    - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
    - <path> – file path on FTP server.
- sftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
    - <ip> – FTP server IP address;
    - <user> – user name (configure the default user name by the *ip sftp client username* command, described in Section ip sftp client username);
    - <password> – password (configure the default password by the *ip sftp client password* command, described in Section ip sftp client password);
    - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
    - <path> – file path on FTP server.
- sftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
    - <ipv6> – FTP server IPv6 address;
    - <interface> – outgoing network interface for link-local addresses;
    - <user> – user name (configure the default user name by the *ip sftp client username* command, described in Section ip sftp client username);
    - <password> – password (configure the default password by the *ip sftp client password* command, described in Section ip sftp client password);
    - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
    - <path> – file path on FTP server.
- scp://[<user>:<password>@]<ip>[<port>]:/<path>
    - <ip> – IP address of server;
    - <user> – user name (configure the default user name by the *ip ssh client username* command, described in Section ip ssh client username);
    - <password> – password (configure the default password by *ip ssh client password* command, described in Section ip ssh client password);
    - <port> – port that listens to TFTP server, separated from IP address by '#' or ':' character;
    - <path> – file path on server.

- scp://[<user>:<password>@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6 address of server;
  - <interface> – outgoing network interface for link-local addresses;
  - <user> – user name (configure the default user name by the *ip ssh client username* command, described in Section ip ssh client username);
  - <password> – password (configure the default password by the *ip ssh client password* command, described in Section ip ssh client password);
  - <port> – port that TFTP server listens to is separated from the IPv6 address by the symbol '#' or ':' (in this case, the IPv6 address must be enclosed in square brackets '[]');
  - <path> – file path on server.

Required privilege level

15

Command mode

CONFIG-ARCHIVE

Example:

```
esr(config-archive)# path tftp://10.10.10.1:/esr-1000/config
```

restore

The command allows to cancel applied but not confirmed configuration and return to the last confirmed one. The command is applied to the whole device configuration. Undoing changes can be carried out only until ‘confirm’ command is entered. When executing ‘restore’ command, there is a loss of unconfirmed configuration.

Syntax

```
restore
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# restore
```

Reversion to the last confirmed configuration is completed.

rollback

The command allows to cancel not applied configuration changes. As a result of command execution, CANDIDATE configuration will be deleted. The command may be used only until ‘commit’ command is entered.

The command is applied to the whole device configuration.

Syntax

```
rollback
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# rollback
```

Cancellation of all not applied configuration changes has been made.

save

The command is used to save CANDIDATE configuration to the device ROM.

Syntax

```
save
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# save
```

Upload of current configuration on the device Flash memory.

show bootvar

The command is used to display information on software images loaded to the device.

Syntax

`show bootvar`

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show bootvar
Image   Version                      Date                    Status         After reboot
-----   -------------------------    --------------------    ------------   ------------
1       1.0.7 build 119[5cd22b8]       date 22/12/2015 time    Not Active
                                       18:00:47
2       1.0.7 build 119[5cd22b8]       date 22/12/2015 time    Active              *
                                       18:00:47
```

show boot-licence

This command is used to view information about the active license of the device used at the stage of loading the secondary bootloader.

Syntax

`show boot-licence`

Parameters

The command does not contain parameters.

Required privilege level

5

Command mode

ROOT

Example:

```
esr# show boot-licence
Licence information
-------------------
Name:    Eltex
Version: 1.0
Type:    ESR-1000
S/N:     NP01000530
MAC:     A8:F9:4B:AA:44:BB
Features:
 KSS - Kaspersky Security System
```

show candidate-config

The command is used to display the device configuration which will be set after the settings are applied (‘commit’ command).

Syntax

```
show candidate-config [ <SECTION> ]
```

Parameters

<SECTION> – configuration section:

- aaa – configuration of authentication, authorization and accounting parameters;
- access-list – access lists configuration;
- bridges – network bridges configuration;
    - [<NUM>] – network bridge number.
- channel-group – channel aggregation group configuration;
- clock – router’s system clock and NTP configuration;
- dhcp – DHCP server, client and Relay agent configuration;
- content-provider – configuration of the source of updates to rules distributed under a commercial license;
- dual-homing – Dual Homing configuration[1];
- extended – extended configuration output;
- hostname – router network name;
- interfaces [<IF>] – interface configuration:
    - <IF> – an interface's name specified in the form described in Section Types and naming order of router interfaces.
- ip-address – IP interface configuration;
- ipv6 [<SUBSECTION>] – IPv6 configuration:
    - address – IPv6 interface configuration;

  - dhcp [<SUBSECTION>] – IPv6 DHCP services configuration:
  - client – IPv6 DHCP server configuration;
    - relay – IPv6 DHCP Relay agent configuration;
    - server – IPv6 DHCP client configuration.
  - routing [<SUBSECTION>] – IPv6 routing configuration:
    - bfd – IPv6 BFD configuration;
    - bgp – IPv6 BGP configuration;
    - ospf – OSPFv3 configuration;
    - prefix-lists – IPv6 prefix lists configuration;
    - rip – RIP configuration;
    - static – static routes configuration.
  - vrrp – IPv6 configuration of VRRP.
- dialplan – dial plan configuration;
- lldp – LLDP configuration;
- mac-address-table – MAC table configuration[1];
- mailservers – configuration of mail servers and domains;
- mdns – mDNS functionality;
- mirroring – mirroring configuration[1];
- mpls – MPLS technology protocols configuration;
- mdns – mDNS functionality configuration;
- multiwan – configuration of redundancy and WAN interface balancing service;
- nat [<SUBSECTION>] – NAT service configuration:
  - source – Source NAT service configuration;
  - destination – Destination NAT service configuration;
  - alg – NAT algorithm configuration.
- netflow – Netflow protocol configuration;
- object-groups [<TYPE> [<NAME>]] – profile configuration;
  - <TYPE> – profile type, may take values:
    - network;
    - address-port;
    - application;
    - mac;
    - service;
    - url.
  - <NAME> – specified type profile name.
- port-security – Port Security configuration[1];
- qos – QoS configuration;
- remote-access [<SUBSECTION>] – L2TP over IPsec profiles and PPTP servers configuration;
  - l2tp – configuration of L2TP over IPsec servers' profiles;
  - pptp – PPTP servers' profile configuration;
  - openvpn – OpenVPN servers' profile configuration;
- remote-client – remote access configuration (SSH, Telnet, etc.);
- rmon – RMON configuration;
- routing [<SUBSECTION>] – routing configuration:
  - bfd – BFD configuration;
  - bgp – BGP configuration;
  - isis – IS-IS protocol configuration;
  - key-chains – authentication keys configuration;
  - ospf – OSPF configuration;
  - prefix-list – prefix list configuration;
  - rip – RIP configuration;
  - route-maps – route maps comfiguration
    - [<RM-NAME>] – route map name.
  - static – static routes configuration;

    - tracking – Tracking objects configuration.
- security [<SUBSECTION>] – IPsec VPN and Firewall services configuration;
    - antispam – "Antispam" service profile configuration;
    - ike – IKE configuration;
    - ipsec – IPsec configuration
    - zone – Firewall zones configuration;
    - zone-pair – configuration of transitions between Firewall zones;
- sip – SIP configuration;
    - profile – SIP profile configuration;
    - service – SIP configuration;
- sflow – sFlow protocol configuration;
- snmp – SNMP server configuration;
- spanning-tree – Spanning Tree Protocol family configuration[1];
- sla – IP SLA service configuration
- system – systemic parameters configuration;
- syslog – Syslog service configuration;
- tunnels [<TYPE> [<NUM>]] – tunnel configuration:
    - <TYPE> – tunnel type, may take values:
        - gre – GRE tunnel configuration;
        - ip4ip4 – configuration of IPv4 over IPv4 tunnels;
        - l2tp – L2TP tunnels configuration;
        - l2tpv3 – L2TPv3 tunnels configuration;
        - lt – logical tunnels configuration;
        - softgre – SoftGRE tunnel configuration[1];
        - vti – VTI tunnels configuration;
        - pptp – PPTP clients configuration;
        - pppoe – PPPoE clients configuration;
        - l2tp – L2TP clients configuration;
        - openvpn – OPENVPN clients configuration;
    - <NUM> – number of the specified tunnel type in the router's configuration.
- vlans – VLAN configuration;
- voice – voice service configuration;
- vrf – VRF configuration;
- vrrp – VRRP protocol configuration;
- wisla – configuration of wiSLA services quality monitoring system;
- wireless-controller – Wi-Fi controller parameters configuration;
- zabbix – Zabbix agent configuration.

[1] In the current firmware version, this functionality is supported only by ESR-1000 router

Required privilege level

10

Command mode

ROOT

Example:

```
esr# show candidate-config
ntp enable
ntp broadcast-client enable
syslog max-files 3
syslog file-size 512
syslog file default info
vlan 2
exit
security zone trusted
exit
security zone untrusted
exit
object-group service telnet
  port-range 23
exit
object-group service ssh
  port-range 22
exit
object-group service dhcp_server
  port-range 67
exit More? Enter – next line; Space – next page; Q – quit; R – show the rest.
```

show configuration changes

This command displays the differences between the configuration files.

Syntax

```
show configuration changes [<CONFIG> < CONFIG>]
```

Parameters

<CONFIG> – configuration file for comparison. May take the following values:

- candidate-config;
- running-config;
- factory-config;
- default-config;
- flash:backup/FILE.

Default value

Without <CONFIG>, the difference between running-config and candidate-config is displayed.

Required privilege level

10

Command mode

ROOT

Example:

```
esr(config-)# show configuration changes
+ interface gigabitethernet 1/0/1.100
+   ip firewall disable
+   ip address 10.54.22.1/24
+ exit
```

show crypto certificates

The command displays information on certificate amount.

Syntax

`show crypto certificates [ <CERTIFICATE-TYPE> ]`

Parameters

<CERTIFICATE-TYPE> – certificate or key type, may take the following values:

- ca – Certificate Authority;
- crl – Certificate Revocation List;
- dh – Diffie-Hellman key;
- server-crt – public server certificate;
- server-key – private server key;
- client-crt – client certificate;
- client-key – client private key;
- ta – HMAC key.

Required privilege level

15

Command mode

ROOT

Example:

```
esr# show crypto certificates
Type           Total
------------   -------
ca                3
dh               1
server key       2
server crt       1
ta               1
crl              1
```

show licence

The command is used to display information on the active device license.

Syntax

```
show licence
```

Parameters

The command does not contain parameters.

Required privilege level

5

Command mode

ROOT

Example:

```
esr# show licence
Licence information
-------------------
Name: X-Telecom
Version: 1.0
Type: ESR-1000
S/N: NP01000046
MAC: A8:F9:4B:AA:03:20
Features:
    DHCP - Dynamic Host Configuration Protocol
    IDS - Empty description
    SWUTIL - View interface's utilization
```

show running-config

The command is used to display the current device configuration.

Syntax

```
show running-config [<SECTION>]
```

Parameters

<SECTION> – configuration section, the description is given in Section show candidate-config.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# show running-config syslog
syslog max-files 3
syslog file-size 512
syslog file default info
syslog console info
```

show storage-devices

The command displays information on connected external storage mediums (USB/MMC memory cards).

Syntax

`show storage-devices { usb | mmc }`

Parameters

usb – USB storage.

mmc – SD/MMC memory card.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show storage-devices mmc
Name                             Total, MB    Used, MB     Free, MB
------------------------------   ----------   ----------   ----------
EF28-D074                        99.79        72.64        27.15
```

show version

The command is used to display the current firmware version and device hardware version.

Syntax

`show version`

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show version
Boot version:
  1.0.7.16 (date 18/11/2015 time 13:40:59)
SW version:
  1.0.7 build 17[d9bdbda] (date 21/11/2015 time 18:06:41)
HW version:
  1v7
```

time-period

The command sets the time period after which the automatic redundancy of the configuration will be performed.

The use of a negative form (no) of the command sets the default value.

Syntax

```
time-period <TIME>
```

```
no time-period
```

Parameters

<TIME> – periodicity of automatic redundancy of the configuration, takes the value in minutes [1..35791394].

Default value

720 minutes

Required privilege level

15

Command mode

CONFIG-ARCHIVE

Example:

```
esr(config-archive)# time-period 1440
```

type

This command sets the backup configuration of the router.

The use of a negative form (no) of the command sets the default mode.

Syntax

```
type <TYPE>

no type
```

Parameters

<TYPE> – type of the router configuration backup. Takes the following values:

- local – saving backup configurations occurs in the flash: backup/section with a file name in the form of 'config_YYYYMMDD_HHMMSS;' Where:
  - YYYY – year, according to the system clock of the router at the time of recording the configuration backup;
  - MM – month, according to the system clock of the router at the time of recording the configuration backup;
  - DD – day, according to the system clock of the router at the time of recording the configuration backup;
  - HH – hour, according to the system clock of the router at the time of recording the configuration backup;
  - MM – minute, according to the system clock of the router at the time of recording the configuration backup;
  - SS – second, according to the system clock of the router at the time of recording the configuration backup.
- remote – backup configurations are saved on a remote server;
- both – backup configurations are saved in section flash:backup and on a remote server;

Default value

remote

Required privilege level

15

Command mode

CONFIG-ARCHIVE

Example:

```
esr(config-archive)# type both
```

7 DNS configuration

- domain ip host
- domain lookup enable
- domain name
- domain name-server

domain ip host

The command defines a static DNS entry.

The use of a negative form (no) of the command removes the entry.

Syntax

```
[no] domain ip host <NAME> <IP>
```

Parameters

<NAME> – host name, set by the string of up to 255 characters.

<IP> – host IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# domain ip host eltex.loc 172.16.0.3
```

domain lookup enable

The command enables DNS names resolution.

The use of a negative form (no) of the command disables DNS names resolution.

Syntax

```
[no] domain lookup enable [ vrf <VRF> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters, within which DNS names resolution will be enabled.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# domain lookup enable
```

domain name

The command allows to assign domain name for the router.

The use of a negative form (no) of the command removes domain name for the router.

Syntax

```
domain name <NAME>
no domain name
```

Parameters

<NAME> – router domain name, set by the string from 1 to 255 characters.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# domain name eltex-co.ru
```

domain name-server

The command defines DNS server IP address used for DNS names resolution.

The use of a negative form (no) of the command removes DNS server address.

Syntax

```
[no] domain name-server <IP> [vrf <VRF>]
```

Parameters

<IP> – IP address of DNS server being used, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the interaction with DNS server will occur.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# domain name-server 172.16.0.1
```

8 mDNS-reflector configuration

- clear ip mdns-reflector
- ip mdns-reflector
- ip mdns-reflector services
- show ip mdns-reflector

clear ip mdns-reflector

This command updates the data in the mDNS-reflector cache

Syntax

```
clear ip mdns-reflector
```

Parameters

The command does not contain parameters.

Default value

None

Required privilege level

1

Command mode

ROOT

Example

```
esr# clear ip mdns-reflector
```

ip mdns-reflector

This command adds an interface to the mDNS-reflector service configuration. To start the service, at least two interfaces must be included in the configuration.

The use of a negative form (no) of the command removes the interface from the mDNS-reflector service configuration.

Syntax

```
ip mdns-reflector
no ip mdns-reflector
```

Parameters

The command does not contain parameters.

Default value

None

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip mdns-reflector
```

ip mdns-reflector services

This command enables mDNS service filtering. The services listed in object-group <NAME> will be allowed, all others will be banned. If the filter list is empty, all services will be available.

The use of a negative form (no) of the command disables mDNS service filtering.

Syntax

```
ip mdns-reflector services <OBJ-GROUP-URL-NAME>
no ip mdns-reflector services
```

Parameters

<OBJ-GROUP-URL-NAME> – url addresses profile name, set by the string of up to 31 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG

Example

```
esr (config)# object-group url test
esr (config-object-group-url)# url _ssh._tcp.local
esr (config-object-group-url)# exit
esr (config)#
esr (config)# ip mdns-reflector services test
```

show ip mdns-reflector

This command outputs a list of services from the mDNS-reflector cache.

Syntax

`show ip mdns-reflector [ interfaces <IF> | service <NAME> ]`

Parameters

<IF> – an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

<NAME> – url addresses profile name, set by the string of up to 31 characters.

Default value

None

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip mdns-reflector
Interface IP address Hostname Service Port
----------- --------------- -------------------- ------------------ ------
br21 172.21.0.10 WEP-12ac.local _brcm-sb._tcp 80
br21 172.21.0.10 WEP-12ac.local _telnet._tcp 23
br21 172.21.0.10 WEP-12ac.local _ssh._tcp 22
br21 172.21.0.10 WEP-12ac.local _http._tcp 80
br21 172.21.0.10 WEP-12ac.local _https._tcp 443
gi1/0/2.20 172.20.0.11 Book-tester-7.local _ssh._tcp 22
gi1/0/2.20 172.20.0.11 Book-tester-7.local _sftp-ssh._tcp 22
gi1/0/2.20 172.20.0.11 Book-tester-7.local _eppc._tcp 3031
gi1/0/2.20 172.20.0.11 Book-tester-7.local _rfb._tcp 5900
gi1/0/2.20 172.20.0.11 Book-tester-7.local _smb._tcp 445
gi1/0/2.20 172.20.0.11 Book-tester-7.local _airplay._tcp 7000
gi1/0/2.20 172.20.0.11 Book-tester-7.local _raop._tcp 5000
gi1/0/2.20 172.20.0.11 Book-tester-7.local _teamviewer._tcp 2020
```

```
esr# show ip mdns-reflector service ssh
Interface IP address Hostname Service Port
----------- --------------- -------------------- ------------------ ------
br21 172.21.0.10 WEP-12ac.local _ssh._tcp 22
gi1/0/2.20 172.20.0.11 Book-tester-7.local _ssh._tcp 22
gi1/0/2.20 172.20.0.11 Book-tester-7.local _sftp-ssh._tcp 22
```

9 General-system parameters configuration

- alias
- boot host auto-config
- hostname
- show cpu network-load
- show cpu history
- show cpu processes
- show cpu utilization
- show system
- show system id
- system fan-speed
- system config-confirm timeout
- system utilization softgre disable

alias

This command configures shortened/specific commands in various command modes.

The use of a negative form (no) of the command removes the entry.

⚠ To activate the configured alias, you should reconnect to the CLI of the router

Syntax

```
alias {<ALIAS_NAME>} {<MODE>} {<COMMAND>}
```

```
no alias <ALIAS_NAME>
```

Parameters

<ALIAS_NAME> – new command name.

<MODE> – names of the command mode in which the new command is present:

- change-expired-password;
- config;
- config-aaa-das-profile;
- config-aaa-radius-profile;
- config-accsess-profile;
- config-acl;
- config-acl-rule;
- config-archive;
- config-bgp;
- config-bgp-af;
- config-bgp-group;
- config-bgp-neighbor;
- config-bridge;
- config-cellular-modem;
- config-cellular-profile;
- config-class-map;
- config-class-policy-map;
- config-das-server;
- config-dhcp-server;
- config-dhcp-vendor-id;
- config-dnat;

- config-dnat-pool;
- config-dnat-rule;
- config-dnat-ruleset;
- config-e1;
- config-gre;
- config-if-gi;
- config-ike-gw;
- config-ike-policy;
- config-ike-proposal;
- config-ip4ip4;
- config-ipsec-policy;
- config-ipsec-proposal;
- config-ipsec-vpn;
- config-ipv6-bgp-af;
- config-ipv6-bgp-group;
- config-ipv6-bgp-neighbor;
- config-ipv6-dhcp-server;
- config-ipv6-ospf;
- config-ipv6-ospf-area;
- config-ipv6-ospf-vlink;
- config-ipv6-pl;
- config-ipv6-wan-rule;
- config-ipv6-wan-target;
- config-ipv6-wan-target-list;
- config-keychain;
- config-keychain-key;
- config-l2tp-server;
- config-l2tpv3;
- config-ldap-server;
- config-line-console;
- config-line-ssh.

<COMMAND> – string of 96 characters long. CLI command. Entered completely, without abbreviations. To enter a command of several words, it is framed by double quotes.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# alias qwe root "show version"
```

boot host auto-config

The command enables the load of router configuration by DHCP Option 150 and 67.

The use of a negative form (no) of the command disables the load of router configuration by DHCP Option 150 and 67.

Syntax

```
[no] boot host auto-config
```

Parameters

The command does not contain parameters.

Required privilege level

10

Default value

Disabled.

Command mode

CONFIG

Example:

```
esr(config)# boot host auto-config
```

hostname

The command allows to assign hostname for the router.

The use of a negative form (no) of the command sets the default router name.

Syntax

```
hostname <NAME>
no hostname
```

Parameters

<NAME> – router host name, set by the string of up to 64 characters.

Required privilege level

10

Default value

esr-10/esr-12v/esr-12vf/esr-14vf/esr-20/esr-21/esr-100/esr-200/esr-1000/esr-1200/esr-1500/esr-1511/esr-1700/esr-3100 – depends on router model.

Command mode

CONFIG

Example:

```
esr(config)# hostname router-1.eltex-co.ru
```

show cpu network-load

The command displays the load generated by network traffic.

Syntax

`show cpu network-load`

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show cpu network-load
CPU ID     CPU load     Heaviest session          Session
                                                  weight
-------    ---------    -----------------------   -------
0          0            --                        0
1          49413        80.88.157.57 ->           9826
                        172.129.22.57
2          46812        80.88.157.75 ->           9895
                        172.129.22.75
3          49229        172.129.22.41 ->          9851
                        80.88.157.41
4          0            --                        0
5          53019        80.88.157.77 ->           9989
                        172.129.22.77
6          39699        80.88.157.79 ->           9863
                        172.129.22.79
7          49726        172.129.22.45 ->          9804
                        80.88.157.45
8          39789        172.129.22.61 ->          9779
                        80.88.157.61
9          36876        80.88.157.59 ->           9775
                        172.129.22.59
10         53041        172.129.22.5 ->           9679
                        80.88.157.5
11         49010        172.129.22.47 ->          9896
                        80.88.157.47
12         53082        172.129.22.13 ->          9650
                        80.88.157.13
13         63027        80.88.157.69 ->           9617
                        172.129.22.69
172        52722        80.88.157.73 ->           10001
                        172.129.22.73
15         55165        80.88.157.71 ->           9924
```

show cpu history

The command displays the history of CPU resources usage.

Syntax

```
show cpu history [ average | max ] [cpu {<CPU>}] [timer {<TIMER>}]
```

Parameters

average | max – specifies the history of which statistics should be displayed, averaged or maximum per interval. Without specifying this key, the history of average statistics is displayed (average)

<CPU> – it is possible to specify the number of a specific CPU by which the history of statistics on CPU resource utilization will be displayed. Without specifying the CPU medium, information will be displayed on all CPUs used by the system.

<TIMER> has the following parameters:

- hours – displays history in the last 72 hours;

- minutes – displays history in the last 60 minutes;
- seconds – displays history in the last 60 seconds;
- When timer is missing, 3 tables containing CPU usage statistics history are displayed

Required privilege level

5

Command mode

ROOT

Example:

```
esr# show cpu history max cpu 0
CPU0
Last 60 seconds:
utilization, %
100
 90    #
 80    #
 70    #
 60    #
 50    ##
 40    ##
 30   # ##
 20   # ##
 10 ############
  ....|....|..
 0 25 50
  time, 5 sec.
Last 60 minutes:
utilization, %
100             #
 90             #
 80             #
 70 #            #
 60 ##           #
 50 ##           #
 40 ##           #
 30 ## #    #    #    #    #    #
 20 ## #    #    #    #    #    #
 10 ############################################################
  ....|....|....|....|....|....|....|....|....|....|....|....|
 0  5 10 15 20 25 30 35 40 45 50 55 60
                           time, min.
Last 72 hours:
utilization, %
100 #       # #           ##
 90 #       # ###           ##
 80 #       # ###           ##
 70 #       #####           ##
 60 #       #####           ##
 50 #       #####           ##
 40 #       #####           ##
 30 #############################################
 20 #############################################
 10 #############################################
  ....|....|....|....|....|....|....|....|....|....|....|....|....|....|..
 0  5 10 15 20 25 30 35 40 45 50 55 60 65 70
                                time, hours
```

show cpu processes

The command displays the use of resources by CPU processes.

Syntax

```
show cpu processes [ active | name <PROCESS_NAME> ]
```

Parameters

active – active processes list;

name – name of a certain process.

Required privilege level

15

Command mode

ROOT

Example:

```
esr# show cpu processes
PID  Name                       CPU 5s    CPU 1m    CPU 5m    Memory    Runtime
---  -------------------------  -------   -------   -------   -------   --------
2013    CLI                      0.00%     0.00%     0.00%     0.40%       --
525    Cp-mgr                    0.00%     0.00%     0.00%     0.08%       --
535    Service-mgr               0.00%     0.00%     0.00%     0.08%       --
532    Env-mgr                   0.00%     0.03%     0.05%     0.11%   14 seconds
530    Alarm-mgr                 0.00%     0.00%     0.00%     0.08%       --
529    Lb                        0.00%     0.00%     0.00%     0.09%       --
528    Session-mgr               0.00%     0.00%     0.00%     0.11%       --
521    Ipc-hub                   0.20%     0.03%     0.04%     0.09%   10 seconds
533    E1                        0.00%     0.00%     0.00%     0.11%       --
522    Syslog-mgr                0.00%     0.00%     0.00%     0.09%       --
524    If-mgr                    0.40%     0.35%     0.34%     0.17%  1 minute and 31 second
534    Systemdb                  0.00%     0.02%     0.02%     0.14%    3 seconds
526    Oi-mgr                    0.40%     0.40%     0.43%     1.14%  1 minute and 57 second
527    Cfgsync-mgr               0.00%     0.00%     0.00%     10.07%    1 second
531    Switch                    0.00%     0.10%     0.11%     1.08%    34 seconds
1877   IPv6 Routing              0.00%     0.02%     0.01%     0.10%     1 second
1885   Routing                   0.00%     0.00%     0.00%     0.10%     1 second
```

show cpu utilization

The command displays the use of CPU resources.

Syntax

```
show cpu utilization
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show cpu utilization
CPU   Last      Last      Last
      5 sec     1 min     5 min
---   -------   -------   -------
0     1.98%     6.75%     20.02%
1     67.50%    15.62%    6.88%
2     65.43%    15.53%    6.94%
3     69.29%    16.08%    7.08%
4     89.90%    20.79%    9.14%
5     74.95%    17.14%    7.49%
6     87.61%    20.18%    8.85%
7     87.41%    20.17%    8.85%
8     81.84%    19.03%    8.40%
9     84.82%    19.79%    8.73%
10    84.53%    19.78%    8.75%
11    83.02%    19.40%    8.58%
12    83.73%    19.55%    8.63%
13    76.56%    16.99%    7.25%
14    70.47%    16.00%    6.95%
15    68.39%    15.07%    6.40%
```

show system

The command displays the device environmental parameters.

Syntax

```
show system
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show system
System type:           Eltex ESR-1000 Service Router
System name:           esr-1000
Software version:      1.0.7 build 118[53264b8] (date 22/12/2015 time 10:23:23)
Hardware version:      1v3
System uptime:         4 minutes and 5 seconds
System MAC address:    A8:F9:4B:AA:03:A0
System serial number:  NP01000050
Main power supply installed:     Present
Main power supply status:        Ok
Reserve power supply installed:  Absent
Fan Level:                        46%
   Fan Table
   ~~~~~~~~~
         Fan 1   Fan 2   Fan 3   Fan 4
------   -----   -----   -----   -----
Status   Ok      Ok      Ok      Ok
   Temperature Table
   ~~~~~~~~~~~~~~~~~
                 CPU        Sensor 1   Sensor 2   Sensor 3
--------------   --------   --------   --------   --------
Temperature, C   63         39         37         49
   Memory Table
   ~~~~~~~~~~~~
        Total, MB          Used, MB           Free, MB
-----   ----------------   ----------------   ----------------
RAM     3798.25            1643.50 (44%)      2154.75 (56%)
FLASH   20.00              1.06 (6%)          18.94 (94%)
```

show system id

The command displays the device serial number.

Syntax

```
show system id
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show system id
Serial number:
  NP01000023
```

system fan-speed

⚠ This command is supported only on ESR-100/200/1000/1200/1500/1511/1700 routers

The command defines the cooling system operating mode.

The use of a negative form (no) of the command sets the default value.

Syntax

`system fan-speed { auto | max }`

`no system fan-speed`

Parameters

auto – automatic control mode;

max – maximum cooling mode.

Default value

ESR-100/200/1500/1511/1700/3100 auto;

ESR-1000/1200 max.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# system fan-speed auto
```

system config-confirm timeout

The command defines the timeout to confirm a current configuration. If the configuration is not confirmed, rollback to a previously applied configuration will occur after timeout.

The use of a negative form (no) of the command sets the default value.

Syntax

```
system config-confirm timeout <TIME>
no system config-confirm timeout
```

Parameters

<TIME> – timeout of configuration redundancy confirmation, takes value in seconds [120..86400].

Default value

600

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# system config-confirm timeout 1200
```

system utilization softgre disable

The command is used to disable counting of SoftGRE tunnels occupancy.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] system utilization softgre disable
```

Parameters

The command does not contain parameters.

Default value

Counting is enabled.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# system utilization softgre disable
```

10 System timer management

- clock timezone
- key
- maxpoll
- minpoll
- ntp access-addresses
- ntp authentication enable
- ntp authentication key-chain
- ntp authentication trusted-key
- ntp broadcast-client enable
- ntp dscp
- ntp enable
- ntp ipv6 source address
- ntp object-group query-only
- ntp object-group serve-only
- ntp peer
- ntp server
- ntp source address
- prefer
- set date
- show date
- show ntp configuration
- show ntp peers
- version

clock timezone

The command sets a timezone.

The use of a negative form (no) of the command sets the default timezone.

Syntax

```
clock timezone <OFFSET>
```

```
no clock timezone
```

Parameters

<OFFSET> – zone designation, including offset in hours regarding Greenwich Mean Time, takes the value of [gmt -12 .. gmt +12].

Default value

gmt 0

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# clock timezone gmt +7
```

key

The command defines a key from the key trust list.

The use of a negative form (no) of the command removes a bind to a specified key.

Syntax

```
key <ID>
no key
```

Parameters

<ID> – key identifier, set in the range of [1..255].

Required privilege level

15

Command mode

CONFIG-NTP

Example:

```
esr(config-ntp)# key 245
```

maxpoll

The command sets the maximum value of time interval between sending messages to NTP server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
maxpoll <INTERVAL>
no maxpoll
```

Parameters

<INTERVAL> – maximum value of poll interval. The command parameter is used as an indicator of the power of two when calculating the interval durability in seconds; it is calculated by raising two to power that is specified by the command parameter, takes the value of [10..17].

Default value

10 (2^{10} = 1024 seconds or 17 minutes 4 seconds)

Required privilege level

10

Command mode

CONFIG-NTP

Example:

```
esr(ntp-remote)# maxpoll 11
```

minpoll

The command sets the minimum value of time interval between sending messages to NTP server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
minpoll <INTERVAL>
no minpoll
```

Parameters

<INTERVAL> – minimum value of poll interval in seconds; it is calculated by raising two to power that is specified by the command parameter, takes the value of [4..6].

Default value

6 (2^{6} = 64 seconds or 1 minutes 4 seconds)

Required privilege level

10

Command mode

CONFIG-NTP

Example:

```
esr(ntp-remote)# minpoll 4
```

ntp access-addresses

This command defines a list of trusted IP addresses with which ntp packets can be exchanged.

The use of a negative form (no) of the command removes a specified list.

Syntax

```
ntp access-addresses <NAME>
no ntp access-addresses
```

Parameters

<NAME> – IP addresses profile name, set by the string of up to 31 characters.

Default value

Filtering disabled.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ntp access-addresses 192.168.16.33
```

ntp authentication enable

The command enables authentication for NTP protocol.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] ntp authentication enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ntp authentication enable
```

ntp authentication key-chain

The command specifies a set of passwords for authentication with a server or peer via md5 hash algorithm.

The use of a negative form (no) of the command removes a bind to the password set.

Syntax

```
ntp authentication key-chain <KEYCHAIN>
no ntp authentication key-chain
```

Parameters

<KEYCHAIN> – key list identifier, set by the string of up to 16 characters.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ntp authentication key-chain lock
```

ntp authentication trusted-key

The command defines a list of trusted keys from the key set.

The use of a negative form (no) of the command removes a specified key.

Syntax

```
[no] ntp authentication trusted-key <ID>
```

Parameters

<ID> – key identifier, set in the range of [1..255].

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ntp authentication trusted-key 25
```

ntp broadcast-client enable

This command enables the mode of receiving broadcast messages from NTP servers for the global configuration and all existing VRFs. The router operates as NTP client. If NTP peers and servers are set in the device configuration, then they are ignored in the broadcast mode.

The use of a negative form (no) of the command disables the broadcast mode.

Syntax

```
[no] ntp broadcast-client enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# ntp broadcast-client enable
```

ntp dscp

The command sets the DSCP code value for the use in IP headers of NTP server outgoing packets.

The use of a negative form (no) of the command sets the default DSCP value.

Syntax

```
ntp dscp <DSCP>
no ntp dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Default value

46

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# ntp dscp 40
```

ntp enable

This command enables the synchronization of the system clock with remote servers via NTP for the global configuration and all created VRFs.

The use of a negative form (no) of the command disables the synchronization via NTP.

Syntax

```
[no] ntp enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# ntp enable
```

ntp ipv6 source address

The command is used to specify source IPv6 address for NTP packets for all peers.

The use of a negative form (no) of the command sets the default source-IPv6 address for NTP packets.

Syntax

`ntp ipv6 source address <IPV6-ADDR>`

`no ntp ipv6 source address`

Parameters

<IPV6-ADDR> – RADIUS server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Default value

IPv6 address of the interface from which NTP packet is sent.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# ntp ipv6 source address fc00::1
```

ntp object-group query-only

The command enables query-only mode that limits interaction via NTP for a certain profile of IP addresses. Only control messages exchange is permitted.

The use of a negative form (no) of the command disables the limitations.

Syntax

```
ntp object-group query-only <NAME>
no ntp object-group query-only
```

Parameters

<NAME> – IP addresses profile name, set by the string of up to 31 characters.

Default value

Work with NTP peers is not limited by anything.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ntp object-group query-only OG-NTP-QO
```

ntp object-group serve-only

The command enables serve-only mode that limits interaction via NTP for a certain profile of IP addresses. You are allowed to respond to NTP requests only; synchronization from other hosts and control messages exchange are prohibited.

The use of a negative form (no) of the command disables the limitations.

Syntax

```
ntp object-group serve-only <NAME>
no ntp object-group serve-only
```

Parameters

<NAME> – IP addresses profile name, set by the string of up to 31 characters.

Default value

Work with NTP peers is not limited by anything

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ntp object-group serve-only OG-NTP-SO
```

ntp peer

The command is used to set partnership relations between NTP servers and to switch to CONFIG-NTP command mode.

NTP server on the router operates in the mode of bi-directional connection with a remote NTP server specified in the command. If one of the partners lost the connection to higher-level NTP server, it will be able to synchronize the time over the partner server.

The use of a negative form (no) of the command removes a specified NTP partner.

Syntax

```
[no] ntp peer { <ADDR> | <IPV6-ADDR> } [ vrf <VRF> ]
```

Parameters

<ADDR> – partner IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – partner IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<VRF> – VRF instance name, set by the string of up to 31 characters, within which an NTP partner will operate.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# ntp peer 10.100.100.1
esr(ntp-remote)#
```

ntp server

The command is used to set partnership relations between NTP servers and to switch to CONFIG-NTP command mode.

The router works with a specified NTP server in one-directional connection mode. In this mode, the router local clock can be synchronized with a remote NTP_ server.

The use of a negative form (no) of the command removes a specified NTP server.

Syntax

`[no] ntp server { <ADDR> | <IPV6-ADDR> } [ vrf <VRF> ]`

Parameters

<ADDR> – server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the NTP server will operate.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# ntp server 10.100.100.2
esr(ntp-remote)#
```

ntp source address

The command is used to specify source IP address for NTP packets for all peers.

The use of a negative form (no) of the command sets the default source-IP address for NTP packets.

Syntax

`ntp source address <ADDR>`

`no ntp source address`

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

Default value

IP address of the interface from which NTP packet is sent.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# ntp source address 10.100.100.2
```

prefer

The command marks this NTP server as a preferable one. All other conditions being equal, this NTP server will be selected for synchronization among all production NTP servers.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] prefer
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-NTP

Example:

```
esr(ntp-remote)# prefer
```

set date

The command sets the system time and date manually.

Syntax

```
set date <TIME> [<DAY> <MONTH> [ <YEAR> ] ]
```

Parameters

<TIME> – system timer, defined as HH:MM:SS, where:

- HH – hours, takes the value of [0..23];
- MM – minutes, takes the value of [0 ..59];
- SS – seconds, takes the value of [0..59];

<DAY> – day of the month, takes values of [1..31];

<MONTH> – month, takes the following values [January/February/March/April/May/June/July/August/September/October/November/December];

<YEAR> – year, takes values of [2001..2037].

Required privilege level

10

Command mode

ROOT

Example:

```
esr# set date 16:35:00 15 May 2014
```

show date

The command displays the current system time and date.

Syntax

```
show date
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show date
Thursday 23:48:33 GMT+7 May 15 2014
```

show ntp configuration

The command displays running NTP configuration.

Syntax

```
show ntp configuration
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# show ntp configuration
 NTP status: Enabled
 NTP mode:   client/server
Address          Type     Prefer   NTP version   Min poll          Max poll
--------------   ------   ------   -----------   ---------------   ---------------
10.100.100.1      peer     yes      NTPv4         6(1m 4s)          11(34m 8s)
10.100.100.2      peer     yes      NTPv4         6(1m 4s)          10(17m 4s)
1.2.3.4           server   no       NTPv4         6(1m 4s)          10(17m 4s)
```

show ntp peers

The command displays the current condition of NTP servers (peers). The tables below contain the list of parameters being displayed and their description.

Table 6 – Remote server condition (peer)

Parameter	Description
remote	DNS name or IP address of server (peer). The first character in the table is used to define server (peer) condition, conditions are described in Table 7.
refid	Binding handle or IP address of who a remote server (peer) is synchronized with. Binding handle types are described in Table 8.
st	Stratum.
t	Relation of the router to a remote server (peer), the types are described in Table 9.

Parameter	Description
when	The time period since the server (peer) was last polled, in seconds ('h' hours, 'd' days).
poll	Polling rate of server (peer).
reach	Eight-bit left-shift register containing polling results (1 = successful, 0 = unsuccessful) is displayed in octal notation.
delay	Packet transmission time - to a server (peer) and back, in milliseconds.
offset	Average constant time offset of the router with respect to the server (peer).
jitter	Average dispersion of time deviation (jitter).

Table 7 – Remote server condition (peer)

Type	Description
space	Indicates that: • there were not responses from a remote server (peer); • the server is not used as the stratum has a big value; • server (peer) uses the router to synchronize its time.
x	server (peer) is not used to synchronize time; it has been dropped by collision algorithm.
-	server (peer) is not used to synchronize time; it has been dropped by cluster algorithm.
#	Remote server (peer) is production but is not used since it was not among first 6 servers (peers) sorted by the synchronization distance; it is a backup server.
+	Production and preferable remote server (peer), enabled by union algorithm.
*	Server (peer) which is currently a primary source of time.

Table 8 – Types of remote server (peer) connection identifier

Type	Description
.ACST.	Manycast server.
.AUTH.	Authentication error.

Type	Description
.AUTO.	Automatic key sequence error.
.BCST.	Broadcast server.
.CRYPT.	Automatic key protocol error.
.DENY.	Server denied access.
.INIT.	Server connection initialization.
.MCST.	Multicast server.
.TIME.	Server connection timeout.
.STEP.	Step change of time, the offset is less than threshold (1000 milliseconds) but more than threshold step (125 milliseconds).
.RATE.	Polling rate exceeding.

Table 9 – Types of relations of router to a remote server (peer)

Type	Description
u	Unicast or manycast client.
b	Broadcast or unicast client.
s	Bi-directional coupling (peer).
A	Manycast server.
B	Broadcast server.
M	Multicast server.

Syntax

```
show ntp peers
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# sh ntp peers
Clock is synchronized, stratum 3, reference is 192.168.1.1
     remote           refid           st t when poll reach  delay   offset  jitter
--------------------- --------------- -- - ---- ---- ----- ------- -------- -------
*192.168.1.1          192.168.1.2      2 u   45  128  377   0.283   0.204   0.049
```

version

The command sets NTP version.

The use of a negative form (no) of the command sets the default value.

Syntax

`version <VERSION>`

`no version`

Parameters

<VERSION> – NTP version, takes values in the range of [1..4].

Default value

4

Required privilege level

15

Command mode

CONFIG-NTP

Example:

```
esr(ntp-remote)# version 3
```

11 AAA configuration

- aaa accounting commands
- aaa accounting login
- aaa authentication attempts max-fail
- aaa authentication enable
- aaa authentication login
- aaa authentication mode
- aaa das-profile
- aaa disable
- aaa radius-profile
- acct-port
- auth-port
- clear users blocked
- clear user-session
- clients
- das-server
- das-server
- dead-interval
- description
- disable
- enable
- enable authentication
- enable password
- exec-timeout
- ip sftp enable
- key
- ldap-server base-dn
- ldap-server bind authenticate root-dn
- ldap-server bind authenticate root-password
- ldap-server bind timeout
- ldap-server dscp
- ldap-server host
- ldap-server naming-attribute
- ldap-server privilege-level-attribute
- ldap-server search filter user-object-class
- ldap-server search scope
- ldap-server search timeout
- line
- login authentication
- password
- port
- priority
- privilege
- privilege
- radius-server dscp
- radius-server host
- radius-server host
- radius-server retransmit
- radius-server timeout
- retransmit
- security passwords default-expired
- security passwords history
- security passwords lifetime
- security passwords lower-case

- security passwords max-length
- security passwords min-length
- security passwords numeric-count
- security passwords special-case
- security passwords symbol-types
- security passwords upper-case
- security snmp-community max-length
- security snmp-community min-length
- show aaa accounting
- show aaa authentication
- show aaa ldap-servers
- show aaa radius-servers
- show aaa tacacs-servers
- show users
- show users accounts
- show users blocked
- source-address
- source-interface
- system configuration-exclusively
- tacacs-server dscp
- tacacs-server host
- tacacs-server timeout
- tech-support login enable
- timeout
- usage
- username

aaa accounting commands

The command configures a list of CLI command accounting methods.

The use of a negative form (no) of the command sets the default value.

Syntax

```
aaa accounting commands stop-only <METHOD>
no aaa accounting commands stop-only
```

Parameters

<METHOD> – accounting methods:

- tacacs – command accounting by TACACS;

Default value

Accounting is not kept.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# aaa accounting commands stop-only tacacs
```

aaa accounting login

The command configures a list of user sessions accounting methods. Accounting is enabled and disabled when a user logs on and disconnects from the system that corresponds to the 'start' and 'stop' values in RADIUS and TACACS messages.

The use of a negative form (no) of the command sets the default value.

Syntax

`aaa accounting login start-stop <METHOD 1> [ <METHOD 2> ]`

`no aaa accounting login start-stop`

Parameters

<METHOD> – accounting methods:

- tacacs – session accounting by TACACS;
- radius – session accounting by RADIUS.

Default value

Session accounting is locally logged.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# aaa accounting login start-stop tacacs
```

aaa authentication attempts max-fail

The command sets the maximum amount of failed authentication attempts until a user is blocked and the blocking time.

The use of a negative form (no) of the command sets the amount of attempts and blocking time by default.

Syntax

```
aaa authentication attempts max-fail <COUNT> <TIME>
no aaa authentication attempts max-fail
```

Parameters

<COUNT> – amount of failed authentication attempts after which a user is blocked, takes the values of [1..65535];

<TIME> – user blocking time in seconds, takes the values of [1..65535].

Default value

Amount of failed attempts – 5

Blocking time – 300

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# aaa authentication attempts max-fail 5 30
```

aaa authentication enable

The command creates the lists of authentication methods for user privileges escalation. If an attempt to authenticate by one method fails, the attempt is made to authenticate by the next method in the list.

The default configuration includes a list named 'default'. The 'default' list includes one authentication method – 'enable'. To use the list for user privileges escalation authentication, it is necessary to bind it by the command described in Section enable authentication.

The use of a negative form (no) of the command removes the authentication methods list.

Syntax

```
aaa authentication enable <NAME> <METHOD 1> [ <METHOD 2> ] [ <METHOD 3> ] [ <METHOD 4> ]
no aaa authentication enable <NAME>
```

Parameters

<NAME> – list name: string of up to 31 characters;

- default – «default» list name.

<METHOD> – authentication methods:

- enable – authentication by enable passwords;
- tacacs – authentication by TACACS;
- radius – authentication by RADIUS;
- ldap – authentication by LDAP.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# aaa authentication enable enable-test tacacs enable
```

aaa authentication login

The command creates the lists of authentication methods for user login. If an attempt to authenticate by one method fails, the attempt is made to authenticate by the next method in the list.

The default configuration includes a list named 'default'; the list contains one authentication method – «local». To use the list for user login authentication, it is necessary to activate it by the command described in Section login authentication.

The use of a negative form (no) of the command removes the authentication methods list.

Syntax

```
aaa authentication login { default | <NAME> } <METHOD 1> [ <METHOD 2> ] [ <METHOD 3> ]
[ <METHOD 4> ]
```

```
no aaa authentication login { default | <NAME> }
```

Parameters

<NAME> – list name, set by the string of up to 31 characters;

Authentication methods:

- local – authentication by local user base;
- tacacs – authentication by TACACS server list;
- radius – authentication by RADIUS server list;
- ldap – authentication by LDAP server list.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# aaa authentication login login-test tacacs local
```

aaa authentication mode

The command defines the mode of operation with authentication method lists.

The use of a negative form (no) of the command removes the authentication methods list.

Syntax

```
[no] aaa authentication mode { break | chain }
```

Parameters

break – during authentication, the following methods will be used in case of a higher priority one being unavailable;

chain – during authentication, the following methods will be used in case of a higher priority one’s refusal.

Default value

chain

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# aaa authentication mode break
```

aaa das-profile

The command is used to add DAS server and to switch to DAS SERVER PROFILE command mode.

The use of a negative form (no) of the command removes a specified profile of dynamic authorization servers (DAS).

Syntax

```
[no] aaa das-profile <NAME>
```

Parameters

<NAME> – DAS profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# aaa das-profile profile1
esr(config-aaa-das-profile)#
```

aaa disable

This command disables access to the router through the console interface.

The use of a negative form of the command (no) enables the access to the router through the console interface.

Syntax

```
[no] aaa disable
```

Parameters

The command does not contain parameters.

Default value

Access to the router via the console interface is enabled.

Required privilege level

10

Command mode

CONFIG-LINE-CONSOLE

Example:

```
esr(config-line-console)# aaa disable
```

aaa radius-profile

The command is used to add RADIUS server profile and to switch to RADIUS SERVER PROFILE command mode.

The use of a negative form (no) of the command removes a specified RADIUS server profile.

Syntax

```
[no] aaa radius-profile <NAME>
```

Parameters

<NAME> – RADIUS server profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# aaa radius-profile profile1
esr(config-aaa-radius-profile)#
```

acct-port

The command specifies a port number to exchange data with a remote RADIUS server when accounting.

The use of a negative form (no) of the command sets the default value.

Syntax

```
acct-port <PORT>
no acct-port
```

Parameters

<PORT> – number of UDP port to exchange data with a remote server, takes values of [1..65535].

Default value

1813

Required privilege level

15

Command mode

CONFIG-RADIUS-SERVER

Example:

```
esr(config-radius-server)# acct-port 4444
```

auth-port

The command specifies a port number to exchange data with a remote RADIUS server when authenticating and authorizing.

The use of a negative form (no) of the command sets the default value.

Syntax

```
auth-port <PORT>
no auth-port
```

Parameters

<PORT> – number of UDP port to exchange data with a remote server, takes values of [1..65535].

Default value

1812

Required privilege level

15

Command mode

CONFIG-RADIUS-SERVER

Example:

```
esr(config-radius-server)# auth-port 4444
```

clear users blocked

The command removes the information on incorrect attempts of various users authentication.

Syntax

```
 clear users blocked <NAME>
```

Parameters

<NAME> – name of the user for which you want to clean the statistics on incorrect authentication attempts, set by the string of up 31 characters.

Without specifying the user name, the whole table of incorrect authentication attempts is cleaned.

Required privilege level

15

Command mode

ROOT

Example:

```
esr# clear users blocked
```

clear user-session

This command closes the CLI user work session.

Syntax

```
clear user-session [ <USERNAME> | <SESSION> ]
```

Parameters

<NAME> – The name of the user whose session should be closed, specified by a string of up to 31 characters.

<SESSION> – number of terminal session to close, set by number in the range [1...10].

Required privilege level

15

Command mode

ROOT

Example:

```
esr# clear users-session
```

clients

The command specifies the list of dynamic authorization clients (DAC) requests of which the dynamic authorization server (DAS) will response to.

The use of a negative form (no) of the command removes the list of dynamic authorization clients (DAC).

Syntax

```
clients object-group <NAME>
no clients
```

Parameters

<NAME> – name of IP addresses profile that contains addresses of dynamic authorization clients, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-DAS-SERVER

Example:

```
esr(config-das-server)# clients object-group pcrf
```

das-server

The command is used to add dynamic authorization server (DAS) and to switch to DAS SERVER command mode. Dynamic authorization servers (DAS) accept RADIUS CoA queries from dynamic authorization clients (DAC), for example disabling or renewed requesting for user services list.

The use of a negative form (no) of the command removes a specified DAS server.

Syntax

[no] das-server <NAME>

Parameters

<NAME> – DAS name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# das-server main
esr(config-das-server)#
```

das-server

The command is used to add dynamic authorization server (DAS) to a configurable profile of dynamic authorization servers.

The use of a negative form (no) of the command removes a specified DAS server.

Syntax

`[no] das-server <NAME>`

Parameters

<NAME> – DAS name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-AAA-DAS-PROFILE

Example

```
esr(config)# das-server mainesr(config-das-server)#
```

dead-interval

The command specifies the interval during which the packets will not be sent to RADIUS server. RADIUS server is switched to this condition when the timeout of waiting for the response to the last valid retry request expires (see Section radius-server retransmit).

The use of a negative form (no) of the command sets the default value.

Syntax

`dead-interval <SEC>`

`no dead-interval`

Parameters

<SEC> – time interval in seconds, takes values of [0..3600].

Default value

120

Required privilege level

10

Command mode

CONFIG-RADIUS-SERVER

Example:

```
esr(config-radius-server)# dead-interval 600
```

description

The command is used to change the description of dynamic authorization servers (DAS) profile or RADIUS servers profile.

The use of a negative form (no) of the command removes a profile description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – profile description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-DAS-SERVER-PROFILE

CONFIG-RADIUS-SERVER-PROFILE

Example:

Set the description for IP addresses profile:

```
esr(config-aaa-das-profile)# description "Main profile"
```

disable

The command reduces the user privilege level to initial one.

Syntax

```
disable
```

Parameters

The command does not contain parameters.

Required privilege level

2

Command mode

ROOT

Example:

```
esr# disable
esr>
```

enable

The command escalates the user privilege level. Methods of user privilege escalation authentication are specified by the command described in Section aaa authentication attempts max-fail.

'Enable' password authentication method is set in the configuration by default. At the same time, passwords are not set, that is, any system user can get the required 15 level of privileges.

To authenticate privilege escalation via TACACS/RADIUS/LDAP, $enab<PRIV>$ users, where <PRIV> – required privilege level of a user to be authenticated, should be created on the server.

Syntax

enable [<PRIV>]

Parameters

<PRIV> – required privilege level, takes value in the range of [2..15].

Default value

15

Required privilege level

1

Command mode

ROOT

Example:

```
esr> enable 10
esr#
```

enable authentication

The command enables the user privilege escalation authentication list that will be used in a configured terminal.

The default configuration includes a list named 'default'; the list contains one authentication method – 'enable'.

The use of a negative form (no) of the command enables the 'default' list.

Syntax

`enable authentication <NAME>`

`no enable authentication`

Parameters

<NAME> – list name, set by the string of up to 31 characters.

Default value

default

Required privilege level

15

Command mode

CONFIG-LINE-CONSOLE

CONFIG-LINE-TELNET

CONFIG-LINE-SSH

Example:

```
esr(config-line-console)# enable authentication enable-test
```

enable password

The command sets the password that will be required when escalating the user privilege level.

> By default, passwords are not set, that is, any system user can get the required 15 level of privileges.

The use of a negative form (no) of the command removes a password from the system.

Syntax

```
enable password { <CLEAR-TEXT> | encrypted <HASH_SHA512> } [ privilege <PRIV> ]
no enable password [ privilege <PRIV> ]
```

Parameters

<CLEAR-TEXT> – password, set by the string of 8 to 32 characters, takes the value of [0-9a-fA-F];

<HASH_SHA512> – hash password via sha512 algorithm, set by the string of 110 characters;

<PRIV> – required privilege level, takes value in the range of [2..15], 15 by default.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# enable password 12345678 privilege 10
```

exec-timeout

The command specifies the time interval after which an idle session will be disconnected.

The use of a negative form (no) of the command sets the default value.

Syntax

```
exec-timeout <SEC>
no exec-timeout
```

Parameters

<SEC> – time interval in minutes, takes values of [1..65535].

Default value

30 minutes

Required privilege level

15

Command mode

CONFIG-LINE-CONSOLE

CONFIG-LINE-SSH

CONFIG-LINE-TELNET

CONFIG-LINE-AUX[1]

Example:

```
esr(config-line-ssh)# exec-timeout 600
```

[1] Only for ESR-21

ip sftp enable

This command enables sftp access on the router for the configurable user.

The use of a negative form of the command (no) disables sftp access for the configurable user.

Syntax

```
[no] ip sftp enable
```

Parameters

None

Default value

Disabled

Required privilege level

10

Command mode

CONFIG-USER

Example:

```
esr(config-user)# ip sftp enable
```

key

The command specifies an authentication password on a remote server.

The use of a negative form (no) of the command removes a specified password for authentication on a remote server.

Syntax

```
key ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }
no key
```

Parameters

<TEXT> – string [8..16] of ASCII symbols (for TACACS server – up to 60 symbols);

<ENCRYPTED-TEXT> – encrypted password, size [8..16] bytes, is specified by the string [16..32] characters (for a TACACS server - up to 120 characters).

Required privilege level

15

Command mode

CONFIG-TACACS-SERVER

CONFIG-RADIUS-SERVER

CONFIG-DAS-SERVER

Example:

```
esr(config-tacacs-server)# key ascii-text 12345678
```

ldap-server base-dn

The command specifies primary DN (Distinguished name) which will be used when searching for users.

The use of a negative form (no) of the command removes a specified primary DN.

Syntax

```
ldap-server base-dn <NAME>
no ldap-server base-dn
```

Parameters

<NAME> – basic DN, set by the string of up to 255 characters.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ldap-server base-dn “dc=example,dc=com”
```

ldap-server bind authenticate root-dn

The command specifies DN (Distinguished name) of a user with administrator rights, under which authorization will take place on LDAP server when searching for users.

The use of a negative form (no) of the command removes a specified user’s DN.

Syntax

```
ldap-server bind authenticate root-dn <NAME>
```

```
no bind authenticate root-dn
```

Parameters

<NAME> – DN of a user with administration rights, set by the string of up to 255 characters.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ldap-server bind authenticate root-dn “cn=admin,dc=example,dc=com”
```

ldap-server bind authenticate root-password

The command specifies password of a user with administrator rights, under which authorization will take place on LDAP server when searching for users.

The use of a negative form (no) of the command removes a specified user’s password.

Syntax

```
ldap-server bind authenticate root-password ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no bind authenticate root-password
```

Parameters

<TEXT> – string [8..16] ASCII characters;

<ENCRYPTED-TEXT> – encrypted password, [8..16] bytes size, set by the string of [16..32] characters.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ldap-server bind authenticate root-password ascii-text 12345678
```

ldap-server bind timeout

The command sets the interval after which the device considers LDAP server as unavailable.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ldap-server bind timeout <SEC>
no ldap-server bind timeout
```

Parameters

<SEC> – time interval in seconds, takes values of [1..30].

Default value

3 seconds

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ldap-server bind timeout 5
```

ldap-server dscp

The command sets the DSCP code value for the use in IP headers of LDAP server outgoing packets.

The use of a negative form (no) of the command sets the default DSCP value.

Syntax

```
ldap-server dscp <DSCP>
no ldap-server dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Default value

63

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# ldap-server dscp 40
```

ldap-server host

The command is used to add LDAP server to the list of servers in use and to switch to LDAP SERVER command mode.

The use of a negative form (no) of the command removes a specified LDAP server.

Syntax

```
 [no] ldap-server host { <ADDR> | <IPV6-ADDR> } [ vrf <VRF> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters.

<ADDR> – LDAP server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – LDAP server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ldap-server host 10.100.100.1
esr(config-ldap-server)#
```

ldap-server naming-attribute

The command sets the name of object attribute, value of which is compared with the name of a desired user on LDAP server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ldap-server naming-attribute <NAME>
no ldap-server naming-attribute
```

Parameters

<NAME> – object attribute name, set by the string of up to 127 characters.

Default value

uid

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ldap-server naming-attribute displayName
```

ldap-server privilege-level-attribute

The command sets the name of object attribute, value of which will define the initial privileges of a user on the device. The attribute should take values of [1..15]. If there is no specified attribute or it contains invalid value, initial user privileges will satisfy privileges of 'remote' user.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ldap-server privilege-level-attribute <NAME>
```

```
no ldap-server privilege-level-attribute
```

Parameters

<NAME> – object attribute name, set by the string of up to 127 characters.

Default value

priv-lvl

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ldap-server privilege-level-attribute title
```

ldap-server search filter user-object-class

The command sets the name of the class of objects among which you should search for users on the LDAP server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ldap-server search filter user-object-class <NAME>
```

```
no ldap-server search filter user-object-class
```

Parameters

<NAME> – object class name, set by the string of up to 127 characters.

Default value

posixAccount

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ldap-server search filter user-object-class shadowAccount
```

ldap-server search scope

The command specifies a user search scope in LDAP server tree.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ldap-server search scope <SCOPE>
no ldap-server search scope
```

Parameters

<SCOPE> – user search scope on LDAP server, takes the following values:

- onelevel – search through the objects on the level following a basic DN tree in LDAP server tree;
- subtree – search through all objects of basic DN subtree in LDAP server tree.

Default value

subtree

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ldap-server search scope onelevel
```

ldap-server search timeout

The command sets the interval after which the device considers that LDAP server has not found users' entries matching the search condition.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ldap-server search timeout <SEC>
no ldap-server search timeout
```

Parameters

<SEC> – time interval in seconds, takes values of [0..30].

Default value

0 – device is waiting for search completion and response from LDAP server.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# ldap-server search timeout 10
```

line

The command performs the switch to a certain terminal configuration mode: local console, remote console (Telnet), remote secure console (SSH).

The use of a negative form (no) of the command sets the default terminal parameters. The default settings are described in sections login authentication and enable authentication.

Syntax

```
[no] line <TYPE>
```

Parameters

<TYPE> – console type:

- console – local console;
- telnet – remote console;
- ssh – secure remote console;

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# line console
esr(config-line-console)#
```

login authentication

The command enables the user login authentication list that will be used in a configured terminal.

The default configuration includes a list named 'default'; the list contains one authentication method – 'local'.

The use of a negative form (no) of the command enables the 'default' list.

Syntax

```
login authentication <NAME>
no login authentication
```

Parameters

<NAME> – list name, set by the string of up to 31 characters.

Default value

default

Required privilege level

15

Command mode

CONFIG-LINE-CONSOLE

CONFIG-LINE-TELNET

CONFIG-LINE-SSH

Example:

```
esr(config-line-console)# login authentication login-test
```

password

The command is used to set a password for a certain user to log-in. The password can be set both in clear text and in the form of sha512 hash.

The use of a negative form (no) of the command removes users password from the system.

Syntax

`password { <CLEAR-TEXT> | encrypted <HASH_SHA512> }`

`no password`

Parameters

<CLEAR-TEXT> – password, set by the string of 8 to 32 characters, takes the value of [0-9a-fA-F];

<HASH_SHA512> – hash password via sha512 algorithm, set by the string of 110 characters.

Required privilege level

15

Command mode

CONFIG-USER

CHANGE-EXPIRED-PASSWORD

Example:

```
esr(config-user) password test
```

port

The command specifies a port number to exchange data with a remote server.

The use of a negative form (no) of the command sets the default value.

Syntax

`port <PORT>`

`no port`

Parameters

<PORT> – number of TCP/UDP port to exchange data with a remote server, takes values of [1..65535].

Default value

49 for TACACS server

389 for LDAP server

Not set for DAS server

Required privilege level

15

Command mode

CONFIG-TACACS-SERVER

CONFIG-LDAP-SERVER

CONFIG-DAS-SERVER

Example:

```
esr(config-tacacs-server)# port 4444
```

priority

The command sets remote server priority. The lower value, the higher the priority of server is.

The use of a negative form (no) of the command sets the default value.

Syntax

```
priority <PRIORITY>
no priority
```

Parameters

<PRIORITY> – remote server priority, takes values in the range of [1..65535].

Default value

1

Required privilege level

15

Command mode

CONFIG-TACACS-SERVER

CONFIG-RADIUS-SERVER

CONFIG-LDAP-SERVER

Example:

```
esr(config-tacacs-server)# priority 5
```

privilege

The command sets the user privilege level. Command set available for a user, depends on the privilege level. Users with 1 to 9 privilege levels only can view the information. Users with 10 to 15 privilege level have access to most part of configuration commands. Users with privilege level 15 have access to the full set of commands. The required command privilege level can be changed, described in the description section.

The use of a negative form (no) of the command sets the default privilege level.

Assignment of initial privilege level to users is as follows:

- Required privilege level is assigned to local database users by the command mentioned above;
- Required privilege level for users authorized via RADIUS is taken from cisco-avpair = "shell:priv-lvl=<PRIV>' attribute;
- Required privilege level for users authorized via TACACS is taken from priv-lvl=<PRIV> attribute;
- privilege level for users authorized by LDAP is taken from the attribute specified by the

privilege-level-attribute command, described in Section line, default is **priv-lvl=<PRIV>**;

If the option mentioned above was not received during user authentication via TACACS/RADIUS/LDAP or an option with invalid value was received, a user will be assigned with 'remote' user privileges, 1 by default; You can change required privilege level of 'remote' user in the same way as for any other user from local base by the command above.

Syntax

```
privilege <PRIV>
```

```
no privilege
```

Parameters

<PRIV> – required privilege level, takes value in the range of [1..15].

Default value

1

Required privilege level

15

Command mode

CONFIG-USER

Example:

```
esr(config-user)# privilege 15
```

privilege

The command sets the minimum privilege level necessary for from a specified command subtree.

The use of a negative form (no) of the command sets the default privilege level.

Syntax

`privilege <COMMAND-MODE> level <PRIV> <COMMAND>`

`no privilege <COMMAND-MODE> <COMMAND>`

Parameters

<COMMAND-MODE> – command mode, the description of modes is given in Table 3;

<PRIV> – required privilege level, takes value in the range of [1..15];

<COMMAND> – command subtree, set by the string of up to 255 characters.

Required privilege level

15

Command mode

CONFIG

Example:

Set the required privilege level 2 for 'show' command subtree of root command mode. The commands of 'show interfaces' subtree should be assigned with privilege level 1.

```
esr(config)# privilege root level 2 "show"
esr(config)# privilege root level 1 "show interfaces"
```

radius-server dscp

The command sets the DSCP code value for the use in IP headers of RADIUS server outgoing packets.

The use of a negative form (no) of the command sets the default DSCP value.

Syntax

`radius-server dscp <DSCP>`

`no radius-server dscp`

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Default value

63

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# radius-server dscp 40
```

radius-server host

The command is used to add RADIUS server to the list of servers in use and to switch to RADIUS SERVER command mode.

The use of a negative form (no) of the command removes a specified RADIUS server.

Syntax

`[no] radius-server host { <ADDR> | <IPV6-ADDR> } [ vrf <VRF> ]`

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters.

<ADDR> – RADIUS server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – RADIUS server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# radius-server host 10.100.100.1
esr(config-radius-server)#
```

radius-server host

The command is used to add RADIUS server to RADIUS server profile.

The use of a negative form (no) of the command removes a specified RADIUS server from the profile.

Syntax

`[no] radius-server host { <ADDR> | <IPV6-ADDR> } [ vrf <VRF> ]`

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters.

<ADDR> – RADIUS server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – RADIUS server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

15

Command mode

CONFIG-RADIUS-SERVER-PROFILE

Example:

```
esr(config-aaa-radius-profile)# radius-server host 10.100.100.1
```

radius-server retransmit

The command sets the number of iterative requests to the last active RADIUS server which will be executed before the execution of requests to RADIUS servers next on the list.

The use of a negative form (no) of the command sets the default value.

Syntax

```
radius-server retransmit <COUNT>
```

```
no radius-server retransmit
```

Parameters

<COUNT> – amount of iterative requests to RADIUS server, takes values of [1..10].

Default value

1

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# radius-server retransmit 5
```

radius-server timeout

The command sets the interval after which the device considers RADIUS server as unavailable.

The use of a negative form (no) of the command sets the default value.

Syntax

```
radius-server timeout <SEC>
no radius-server timeout
```

Parameters

<SEC> – time interval in seconds, takes values of [1..30].

Default value

3 seconds

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# radius-server timeout 5
```

retransmit

The command sets the number of iterative requests to RADIUS server which will be executed before the execution of requests to RADIUS servers next on the list.

The use of a negative form (no) of the command sets the default value.

Syntax

```
retransmit <COUNT>
no retransmit
```

Parameters

<COUNT> – amount of iterative requests to RADIUS server, takes values of [1..10].

Default value

Is not specified, global parameter value described in Section radius-server retransmit is used.

Required privilege level

15

Command mode

CONFIG-RADIUS-SERVER

Example:

```
esr(config)# retransmit 5
```

security passwords default-expired

The command enables the default password reset request for admin user.

The use of a negative form (no) of the command disables the default password reset request.

Syntax

```
[no] security passwords default-expired
```

Parameters

The command does not contain parameters

Default value

Password reset request is disabled by default.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# security passwords default-expired
```

security passwords history

The command enables prohibition mode for the use of previously set local user passwords. Amount of passwords kept in the router's memory is specified as a parameter.

The use of a negative form (no) of the command sets the default value.

Syntax

```
security passwords history <COUNT>
```

```
no security passwords history
```

Parameters

<COUNT> – amount of passwords kept in the router's memory [0..15]. When reducing this value, the extra older passwords are deleted.

Default value

1

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# security passwords history 5
```

security passwords lifetime

The command sets local user password lifetime. When attempting to connect a user with an expired password, the user will be switched to the mode of forced password reset.

The use of a negative form (no) of the command sets the default value.

Syntax

```
security passwords lifetime <TIME>
```

```
no security passwords lifetime
```

Parameters

<TIME> – interval of password lifetime in days, takes values of [1..365].

Default value

The lifetime of local user password is unlimited.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# security passwords lifetime 30
```

security passwords lower-case

The command sets the minimum amount of lower case letters in local user password, ENABLE password, SMNPv3 user name and SMNPv1/SMNPv2 with community.

The use of a negative form (no) of the command sets the default value.

Syntax

```
security passwords lower-case <COUNT>
no security passwords lower-case
```

Parameters

<COUNT> – minimum amount of lower case letters in local user password, ENABLE password, SMNPv3 user name and SMNPv1/SMNPv2 with community [0..32].

Default value

0

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# security passwords lower-case 2
```

security passwords max-length

This command sets a limit on the maximum length of the local user password and the ENABLE password.

The use of a negative form (no) of the command sets the default value.

Syntax

```
security passwords max-length <NUM>
no security passwords max-length
```

Parameters

<NUM> – maximum amount of characters in password, set in the range of [8..32].

Default value

128

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# security passwords max-length 30
```

security passwords min-length

This command sets a limit on the minimum length of the local user password and the ENABLE password.

The use of a negative form (no) of the command sets the default value.

Syntax

```
security passwords min-length <NUM>
no security passwords min-length
```

Parameters

<NUM> – minimum amount of characters in password, set in the range of [8..32].

Default value

8

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# security passwords min-length 10
```

security passwords numeric-count

The command sets the minimum amount of digits in local user password, ENABLE password, SMNPv3 user name and SMNPv1/SMNPv2 with community.

The use of a negative form (no) of the command sets the default value.

Syntax

```
security passwords numeric-count <COUNT>
no security passwords numeric-count
```

Parameters

<COUNT> – minimum amount of digits in local user password, ENABLE password, SMNPv3 user name and SMNPv1/SMNPv2 with community [0..32].

Default value

0

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# security passwords numeric-count 2
```

security passwords special-case

The command sets the minimum amount of special characters in local user password, ENABLE password, SMNPv3 user name and SMNPv1/SMNPv2 with community.

The use of a negative form (no) of the command sets the default value.

Syntax

```
security passwords special-case <COUNT>
```

```
no security passwords special-case
```

Parameters

<COUNT> – minimum amount of special characters in local user password, ENABLE password, SMNPv3 user name and SMNPv1/SMNPv2 with community [0..32].

Default value

0

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# security passwords special-case 2
```

security passwords symbol-types

The command sets the minimum amount of special characters in local user password, ENABLE password, SMNPv3 user name and SMNPv1/SMNPv2 with community.

The use of a negative form (no) of the command sets the default value.

Syntax

```
security passwords symbol-types <COUNT>
```

```
no security passwords symbol-types
```

Parameters

<COUNT> – minimum amount of special characters in local user password, ENABLE password, SMNPv3 user name and SMNPv1/SMNPv2 with community [1..4].

Default value

1

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# security passwords symbol-types 2
```

security passwords upper-case

The command sets the minimum amount of upper case letters in local user password, ENABLE password, SMNPv3 user name and SMNPv1/SMNPv2c with community.

The use of a negative form (no) of the command sets the default value.

Syntax

```
security passwords upper-case <COUNT>
no security passwords upper-case
```

Parameters

<COUNT> – minimum amount of upper case letters in local user password, ENABLE password, SMNPv3 user name and SMNPv1/SMNPv2 with community [0..32].

Default value

0

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# security passwords upper-case 2
```

security snmp-community max-length

This command is used to set a limit on the maximum length of SMNPv1/SMNPv2 from a community.

The use of a negative form (no) of the command sets the default value.

Syntax

```
security snmp-community max-length <NUM>
no security snmp-community max-length
```

Parameters

<NUM> – maximum amount of characters in comunity, set in the range of [1..128].

Default value

128

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# security snmp-community max-length 30
```

security snmp-community min-length

This command is used to set a limit on the minimum length of SMNPv1/SMNPv2 from a community.

The use of a negative form (no) of the command sets the default value.

Syntax

```
security passwords min-length <NUM>
no security passwords min-length
```

Parameters

<NUM> – minimum amount of characters in comunity, set in the range of [1..128].

Default value

1

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# security snmp-community min-length 10
```

show aaa accounting

The command displays configured accounting parameters.

Syntax

`show aaa accounting`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# show aaa accounting
Login :           radius
Commands :        tacacs
```

show aaa authentication

The command displays lists of user authentication methods, as well as active lists of each type of terminals.

Syntax

`show aaa authentication`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# show aaa authentication
   Login Authentication Method Lists
   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
List                Methods
----------------    --------------------------------
default             local
   Enable Authentication Method Lists
   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
List                Methods
----------------    --------------------------------
default             enable
   Lines configuration
   ~~~~~~~~~~~~~~~~~~~
Line        Login method list                  Enable method list
---------   --------------------------------   --------------------------------
console     default                            default
telnet      default                            default
ssh         default                            default
```

show aaa ldap-servers

The command displays LDAP servers parameters.

Syntax

```
show aaa ldap-servers
```

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

ROOT

Example:

```
esr# show aaa ldap-servers
Base DN:                      dc=example,dc=com
Root DN:                      cn=admin,dc=example,dc=com
Root password:                CDE65039E5591FA3
Naming attribute:             uid
Privilege level attribute:    priv-lvl
User object class:            posixAccount
DSCP:                         63
Bind timeout:                 3
Search timeout:               0
Search scope:                 subtree
IP Address                         Port           Priority
--------------------------------   ------------   ------------
10.100.100.1                       389            1
```

show aaa radius-servers

The command displays RADIUS servers parameters.

Syntax

```
show aaa radius-servers
```

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

ROOT

Example:

```
esr# show aaa radius-servers
Timeout:     3
Retransmit:  1
DSCP:        63
IP Addres        Timeout      Priority     Usage        Key
------------    ----------   ----------   ----------   ---------------------------
2.2.2.2             --           1            all          9DA7076CA30B5FFE0DC9C4
2.4.4.4             --           1            all          9DA7076BA30B4EFCE5
```

show aaa tacacs-servers

The command displays TACACS servers parameters.

Syntax

```
show aaa tacacs-servers
```

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

ROOT

Example:

```
esr# show aaa tacacs-servers
Timeout :       3
DSCP:          63
IP Address               Port           Priority       Key
----------------------   ------------   ------------   --------------------------------
10.100.100.1             49             1              CDE65039E5591FA3
10.100.100.5             49             10             CDE65039E5591FA3
```

show users

The command displays system users active sessions.

Syntax

```
show users
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show users
User name           Logged in at         Host              Timers Login/Priv    level
--------------      -----------------    --------------    -----------------    -----
admin               13/02/15 01:14:25    Console           00:29:57/00:00:00    15
1 user sessions.
```

show users accounts

The command displays system users configuration.

Syntax

`show users accounts`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# show user accounts
Name                                   Password                               Privilege
--------------------------------       --------------------------------       ---------
admin                                  $6$1sxrvGaV8Za8oX/K$YNel5xYPZ4cj       15
                                       bemYWYNpQBQKDxWE9v0aoKgQ
                                       kRCEb0EMNuusO9Kmg7UBs7nA3buEM87e
                                       Eu.rA6tZq0
techsupport                            $6$YfwntIwU$ah7UxPZTemKhjpSWvVsV       15
                                       9jHcp. 9lweQaSldw7ZtUr
                                       uH66uZx9.EBASff//hUj8ObUaC484TNR
                                       x.
remote                                 $6$YfwntIwU$ah7UxPZTemKhjpSWvVsV       1
                                       9jHcp.kqFAK.vmvyY9lweQaSldw7ZtUr
                                       uH66uZx9.EBASff//hUj8ObUaC484TNR
                                       x.
operator                               $6$eILpbbyRxedCzvVD$4RHP08mjXvNf       1
                                       urX7V/ULCZ1oHIWMwE6h5f
                                       zgwZQUZcPoZCEyaqQQqCicRMRuPwhxrQ
                                       bvGChWreW1
```

show users blocked

The command displays the list of users with incorrect password entered. A user is removed from the list after entering the correct password during authentication.

Syntax

```
show users blocked [ <NAME> ]
```

Parameters

<NAME> – name of the user for which you want to show the statistics on incorrect authentication attempts, set by the string of up 31 characters.

Without specifying the user name, the whole table of incorrect authentication attempts is shown.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show users blocked
User name                Failures    Latest failure      From
---------------------    ---------   -----------------   ----------------
tester                   4           10/09/17 08:29:42   0.0.0.0
```

source-address

The command specifies IPv4/IPv6 address of the router which will be used as IPv4/IPv6 source address in packets sent to AAA server being configured.

The use of a negative form (no) of the command removes a specified source IPv4/IPv6 address.

Syntax

```
source-address { <ADDR> | <IPV6-ADDR> }
```

```
no source-address
```

Parameters

<ADDR> – source IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – source IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

15

Command mode

CONFIG-RADIUS-SERVER

CONFIG-TACACS-SERVER

CONFIG-LDAP-SERVER

Example:

```
esr(config-radius-server)# source-address 220::71
```

source-interface

The command specifies router interface or tunnel, IPv4/IPv6 address of which will be used as IPv4/IPv6 source address in packets sent to AAA server being configured.

The use of a negative form (no) of the command removes a specified interface or tunnel.

Syntax

```
source-interface { <IF> | <TUN> }
no source-interface
```

Parameters

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

Required privilege level

15

Command mode

CONFIG-RADIUS-SERVER

Example:

```
esr(config-radius-server)# source-interface gigabitethernet 1/0/1
```

system configuration-exclusively

This command is used to limit the number of CLI sessions to one.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] system configuration-exclusively
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# system configuration-exclusively
```

tacacs-server dscp

The command sets the DSCP code value for the use in IP headers of TACACS server outgoing packets.

The use of a negative form (no) of the command sets the default DSCP value.

Syntax

```
tacacs-server dscp <DSCP>
no tacacs-server dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Default value

63

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# tacacs-server dscp 40
```

tacacs-server host

The command is used to add TACACS server to the list of servers in use and to switch to TACACS SERVER command mode.

The use of a negative form (no) of the command removes a specified TACACS server.

Syntax

`[no] tacacs-server host { <ADDR> | <IPV6-ADDR> } [ vrf <VRF> ]`

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters.

<ADDR> – TACACS server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<IPV6-ADDR> – TACACS server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# tacacs-server host 10.100.100.1
esr(config-tacacs-server)#
```

tacacs-server timeout

The command sets the interval after which the device considers TACACS server as unavailable.

The use of a negative form (no) of the command sets the default value.

Syntax

`tacacs-server timeout <SEC>`

`no tacacs-server timeout`

Parameters

<SEC> – time interval in seconds, takes values of [1..30].

Default value

3 seconds.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# tacacs-server timeout 5
```

tech-support login enable

The command enables low-level remote access to the system using the 'techsupport' user. Low-level access to the system provides technical support with all required information when it is necessary.

The use of a negative form (no) of the command disables low-level remote access to the system using the 'techsupport' user.

Syntax

```
[no] tech-support login enable
```

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# tech-support login enable
```

timeout

The command sets the interval after which the device considers RADIUS server as unavailable.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timeout <SEC>
no timeout
```

Parameters

<SEC> – time interval in seconds, takes values of [1..30].

Default value

Is not specified, global timer value described in Section radius-server timeout is used.

Required privilege level

10

Command mode

CONFIG-RADIUS-SERVER

Example:

```
esr(config-radius-server)# timeout 7
```

usage

The command specifies type of connections for authentication of which RADIUS server will be used.

The use of a negative form (no) of the command sets the default value.

Syntax

```
usage { all | aaa | auth | acct | pptp | l2tp }
no usage
```

Parameters

all – all connection types;

aaa – RADIUS server will be used for authentication, authorization and accounting of telnet, ssh console sessions;

auth – RADIUS server will be used for authentication and authorization of telnet, ssh console sessions;

acct – RADIUS server will be used for accounting of telnet, ssh console sessions;

pptp – RADIUS server will be used for authentication, authorization and accounting of remote users connected via PPTP;

l2tp – RADIUS server will be used for authentication, authorization and accounting of remote users connected via L2TP over IPsec.

Default value

all

Required privilege level

15

Command mode

CONFIG-RADIUS-SERVER

Example:

```
esr(config-radius-server)# usage pptp
```

username

The command adds a user to the local user base and performs the switch to user parameters configuration mode.

The use of a negative form (no) of the command removes a user from the system.

Syntax

```
[no] username <NAME>
```

Parameters

<NAME> – user name, set by the string of up to 31 characters. If the command is used for removal, when specifying the 'all' value all users will be removed.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# username test
esr(config-user)#
```

12 Interface monitoring and configuration

- General commands
  - clear interfaces counters
  - description
  - history statistics
  - interface
  - ip tcp adjust-mss
  - load-average
  - mode
  - mtu
  - rate-limit arp-broadcast
  - rate-limit arp-broadcast pps
  - show interfaces counters
  - show interfaces description
  - show interfaces history
  - show interfaces status
  - show interfaces utilization
  - shutdown
  - show system jumbo-frames
  - speed
  - switchport dot1q ethertype egress stag
  - switchport mode
  - system jumbo-frames
- Physical interfaces
  - snmp trap link-status
  - show interfaces protected-ports
  - show interfaces sfp
  - show interfaces switch-port configuration
  - show interfaces switch-port status
  - threshold rx-utilization
  - threshold rx-utilization monitoring
  - threshold tx-utilization
  - threshold tx-utilization monitoring
  - switchport community
  - switchport protected
  - switchport protected-port
- Aggregated interfaces
  - channel-group
  - lacp port-priority
  - lacp system-priority
  - lacp timeout
  - port-channel load-balance
  - show interfaces port-channel
  - show lacp counters
  - show lacp interfaces
  - show lacp parameters
- E1/multilink
  - enable
  - ip tcp header-compression
  - ip tcp compression-connections
  - mrru
  - password
  - ppp authentication chap

  - ppp chap hostname
  - ppp chap password
  - ppp chap refuse
  - ppp chap username
  - ppp ipcp accept-address
  - ppp ipcp remote-address
  - ppp max-configure
  - ppp max-failure
  - ppp max-terminate
  - ppp mru
  - ppp multilink
  - ppp multilink-group
  - ppp timeout keepalive
  - ppp timeout retry
  - show controllers e1
  - switchport e1 slot
  - switchport e1 clock source
  - switchport e1 crc
  - switchport e1 framing
  - switchport e1 invert data
  - switchport e1 linecode
  - switchport e1 timeslots
  - switchport e1 unframed
  - switchport mode e1
- Serial interfaces
  - chat-script
  - clear line aux
  - databits
  - dialer
  - dialer idle-timeout
  - dialer in-band
  - dialer map
  - dialer string
  - flowcontrol
  - line aux
  - modem inout
  - parity
  - show line aux
  - speed
  - stopbits
  - transport telnet port
- Wireless modems
  - allowed-auth
  - allowed-mode
  - apn
  - cellular modem
  - cellular profile
  - device
  - enable
  - ip-version
  - mru
  - number
  - password
  - pin

  - preferred-mode
  - profile
  - show cellular configuration modem
  - show cellular configuration profile
  - show cellulas status modem
  - user
- Phone port configuration
  - authentication name
  - authentication password
  - call-forwarding busy
  - call-forwarding busy number
  - call-forwarding no-answer
  - call-forwarding no-answer number
  - call-forwarding no-answer timeout
  - call-forwarding unconditional
  - call-forwarding unconditional number
  - call-waiting
  - caller-id mode
  - dial-type
  - enable
  - flash call-transfer
  - flash call-transfer mode
  - flash timer
  - hotline
  - hotline ipt
  - hotline pstn
  - hotline number
  - hotline number ipt
  - hotline number pstn
  - hotline timeout
  - hotline timeout ipt
  - hotline timeout pstn
  - hybrid rx
  - hybrid tx
  - ipt prefix-name
  - ipt prefix-number
  - ipt offhook-ringing
  - ipt ring-number
  - profile sip
  - profile pbx
  - pstn transmit-number
  - pstn transmit-prefix
  - shutdown
  - sip port
  - sip user display-name
  - sip user phone
  - timing delay
  - timing digit
  - timing flash
  - timing pulse-digit
  - timing pulse-interdigit
  - timing pulse-pause
  - timing tone-digit
  - timing tone-interdigit

- Phone port testing
    - test voice-port start
    - test voice-port status

Naming order of router interfaces is described in Section Types and naming order of router interfaces.

Commands entered in interface (interface group) configuration mode are applied to a selected interface (interface group).

General commands

clear interfaces counters

The command performs the reset of specified system interface/interface group counter.

Syntax

```
clear interfaces counters [<IF>]
```

Parameters

<IF> – name of an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

You can specify several interfaces separated by commas ',' or you can specify the range of interfaces with '-'. If interface indexes are not specified, then the counters of all interfaces of a specified group will be cleared. When executing a command without parameters, all system interface counters will be cleared.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# clear interfaces counters gigabitethernet 1/0/5
```

description

The command is used to change a description of configured interface.

The use of a negative form (no) of the command removes a specified description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – interface description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-E1

CONFIG-MULTILINK

CONFIG-SERVICE-PORT

CONFIG-CELLULAR-PROFILE

CONFIG-CELLULAR-MODEM

CONFIG-VOICE-PORT

CONFIG-VOIP-PROFILE

CONFIG-LINE-AUX[1]

Example:

```
esr(config-if-gi)# description "Uplink interface"
```

[1] Only for ESR-21

history statistics

The command enables maintenance of a current interface usage statistics.

The use of a negative (no) form of the command disable maintenance of a current interface usage statistics.

Syntax

```
[no] history statistics
```

Parameters

None.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-MULTILINK

CONFIG-E1

CONFIG-BRIDGE

Example:

```
esr:esr(config-if-gi)# history statistics
esr:esr(config-if-gi)#
```

interface

The command allows you to switch to the configuration mode of one or more interfaces.

The use of a negative form (no) of the command returns the default interface configuration.

Syntax

```
[no] interface <IF>
```

Parameters

<IF> – name of an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

Required privilege level

10

Command mode

CONFIG

Example 1

Switch to Ethernet interface gi 1/0/20 configuration mode:

```
esr(config)# interface gigabitethernet 1/0/20
esr(config-if-gi)#
```

Example 2

Switch to Ethernet interface gi 1/0/2 configuration mode:

```
esr(config)# interface tengigabitethernet 1/0/2
esr(config-if-te)#
```

Example 3

Switch to a virtual interface configuration mode:

```
esr(config)# interface loopback 5
esr(config-loopback)#
```

Example 4

Switch to a sub interface configuration mode:

```
esr(config)# interface gigabitethernet 1/0/20.20
esr(config-subif)#
```

Example 5

Switch to the port-channel 2 configuration mode:

```
esr(config)# interface port-channel 2
esr(config-port-channel)#
```

Example 6

Switch to interface e1 1/0/1 configuration mode:

```
esr(config)# interface e1 1/0/1
esr(config-e1)#
```

Example 7

Switch to multilink 1 configuration mode:

```
esr(config)# interface multilink 1
esr(config-multilink)#
```

ip tcp adjust-mss

This command overrides the value of the MSS (Maximum segment size) field in incoming TCP packets.

The use of a negative form (no) of the command disables MSS field value correction.

Syntax

```
ip tcp adjust-mss <MSS>
no ip tcp adjust-mss
```

Parameters

<MSS> – MSS value, takes values in the range of [500..1460].

Default value

1460

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

Example

```
esr(config-if-gi)# ip tcp adjust-mss 1400
```

load-average

The command sets a time interval used for interface utilization calculation.

The use of a negative form (no) of the command sets the default value.

Syntax

```
load-average <TIME>
no load-average
```

Parameters

<TIME> – interval in seconds, takes values of [5..150].

Default value

5

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-E1

CONFIG-MULTILINK

CONFIG-BRIDGE

CONFIG-LOOPBACK

Example:

```
esr(config-if-gi)# load-average 30
```

mode

The command sets an operation mode of physical/aggregated interface.

The use of a negative form (no) of the command sets the default value.

Syntax

```
mode <MODE>
no mode
```

Parameters

<MODE> – physical/aggregated interface operation mode taking the following values:

- switchport – sets L2 mode. Vlan permitting is allowed, IP address assigning and sub/qinq interfaces creating are not.
- routerport – sets L3 mode. IP address assigning and sub/qinq interfaces creating are allowed, vlan permitting on an interface is not.
- hybrid – vlan permitting, IP address assigning and sub/qinq interfaces creating are allowed[1].

Default value

routerport

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example:

```
esr(config-if-gi)# mode switchport
```

[1] Available only for ESR-1000/1200/1500/1511/1700 routers.

mtu

The command specifies MTU (Maximum Transmission Unit) size for interfaces.

The use of a negative form (no) of the command sets the default MTU value.

Syntax

```
mtu <MTU>
no mtu
```

Parameters

<MTU> – MTU value in bytes, takes values in the range of:

ESR-10/12V/12VF/14VF – [552..9600]

ESR-20/21 – [552..9500]

ESR-100/200/1000/1200/1500/1511/1700 – [552..10000]

ESR-3100 – [552..9190]

For Serial, E1 and multilink interfaces – [552-1500] for all router models.

MTU value above 1500 can be set only with enabled Jumbo frame support described in Section system jumbo-frames.

Default value

1500

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-CELLULAR-MODEM

Example:

```
esr(config-if-gi)# mtu 1400
```

rate-limit arp-broadcast

The command enables limiting of arp requests receiving on a bridge interface.

When the negative (no) form of the command is used, limiting of arp requests receiving on a bridge interface is disabled.

Syntax

```
[no] rate-limit arp-broadcast
```

Parameters

None.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-BRIDGE

Example:

```
esr(config-bridge)# rate-limit arp-broadcast
```

rate-limit arp-broadcast pps

The command sets the maximum limit for arp-requests on a bridge interface. The parameter is available only if the limiting of arp requests receiving on a bridge interface is enabled

The use of a negative form (no) of the command sets the default value.

Syntax

```
rate-limit arp-broadcast pps <PPS>
no rate-limit arp-broadcast pps
```

Parameters

<PPS> – number of arp-requests in the range of [1..65535].

Default value

100

Required privilege level

10

Command mode

CONFIG-BRIDGE

Example:

```
esr(config-bridge)# rate-limit arp-broadcast pps 2000
```

show interfaces counters

The command displays system interface counters: ports, sub interfaces, aggregation groups, network bridges.

Syntax

```
show interfaces counters [ <IF> ]
```

Parameters

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces.

The command displays counters for router ports, subinterfaces and tunnel interfaces.

You may specify several interfaces. If interface indexes are not specified, then the counters of all interfaces of a specified group will be shown. If a certain interface is specified, the detailed information on this interface will be displayed. When executing a command without parameter, all system interface counters will be displayed.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show interfaces counters gigabitethernet 1/0/4-6
Interface       UC recv         Bytes recv      Errors recv     MC recv
-------------   -------------   -------------   -------------   -------------
gi1/0/4         0               0               0               0
gi1/0/5         0               0               0               0
gi1/0/6         0               0               0               0
Interface       UC sent         Bytes sent      Errors sent
-------------   -------------   -------------   -------------
gi1/0/4         0               0               0
gi1/0/5         1138            393748          0
gi1/0/6         0               0               0
esr# show interfaces counters gigabitethernet 1/0/4
 Packets received:                0
 Bytes received:                  0
 Dropped on receive:              0
 Receive errors:                  0
 Multicasts received:             0
 Receive length errors:           0
 Receive buffer overflow errors:  0
 Receive CRC errors:              0
 Receive frame errors:            0
 Receive FIFO errors:             0
 Receive missed errors:           0
 Receive compressed:              0
 Packets transmitted:             0
 Bytes transmitted:               0
 Dropped on transmit:             0
 Transmit errors:                 0
 Transmit aborted errors:         0
 Transmit carrier errors:         0
 Transmit FIFO errors:            0
 Transmit heartbeat errors:       0
 Transmit window errors:          0
 Transmit comressed:              0
 Collisions:                      0
```

show interfaces description

The command is used to display system interfaces description.

Syntax

```
show interfaces description [ <IF> ]
```

Parameters

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces.

You may specify several interfaces in the command. If interface indexes are not specified, then the statuses of all interfaces of a specified group will be shown. When executing a command without parameter, all system interface descriptions will be displayed.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show interfaces description gigabitethernet 1/0/4-5
Interface       Admin   Link    Description
                State   State
-------------   -----   -----   ----------------------------------------------------
gi1/0/4         Up      Down    Link to NSK
gi1/0/5         Up      Down    Link to MSK
```

show interfaces history

The command is used to view interface usage statistics.

Syntax

`show interfaces history [<IF>] [timer <TIMER>]`

Parameters

<IF> – name of an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

<TIMER > – timer optional key. Has the following parameters:

- hours displays history in the last 72 hours
- minutes displays history in the last 60 hours
- seconds displays history in the last 60 seconds
- When timer is missing, 3 tables containing tunnel usage history are displayed

Required privilege level

5

Command mode

ROOT

Example:

```
esr# show interfaces history gi 1/0/1 timer minutes
gi1/0/1
Last 60 minutes:
Timer  Recv utilization, Kbit/s  Sent utilization, Kbit/s  Recv errors  Sent errors  Output
drops
------ ------------------------- ------------------------- ------------ ------------
--------------
0-1   240            16             0       0       0
1-2   961            64             0       0       0
2-3   962            64             0       0       0
3-4   962            64             0       0       0
4-5   960            64             0       0       0
5-6   961            64             0       0       0
6-7   719            64             0       0       0
7-8   960            64             0       0       0
8-9   800            65             0       0       0
9-10  962            64             0       0       0
10-11 865            64             0       0       0
11-12 962            64             0       0       0
12-13 817            65             0       0       0
13-14 962            65             0       0       0
14-15 961            65             0       0       0
15-16 880            60             0       0       0
16-17 960            63             0       0       0
17-18  0             0              0       0       0
18-19  0             0              0       0       0
19-20  0             0              0       0       0
20-21  0             0              0       0       0
21-22  0             0              0       0       0
```

show interfaces status

The command is used to display system interfaces status.

Syntax

```
show interfaces status [<IF>]
```

Parameters

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces.

You may specify several interfaces in the command. If interface indexes are not specified, then the statuses of all interfaces of a specified group will be shown. If a certain interface is specified, the detailed information on this interface will be displayed. When executing a command without parameter, all system interface statuses will be displayed.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show interfaces status gigabitethernet 1/0/1-2
Interface       Admin   Link    MTU      MAC address          Uptime
                state   state
-------------   -----   -----   ------   ------------------   -------------------------
gi1/0/1         Up      Down    1500     a8:f9:4b:aa:53:fc    --
gi1/0/2         Up      Up      1500     a8:f9:4b:aa:53:fd    15 hours, 17 minutes and 52
 seconds
```

show interfaces utilization

The command is used to display the current load on physical interfaces.

Syntax

```
show interfaces utilization [ <IF> ]
```

Parameters

<IF> – a physical interface's name specified in the form described in Section Types and naming order of router interfaces.

You can specify several interfaces separated by commas ',' or you can specify the range of interfaces with '-'. If interface indexes are not specified, then the current load of all interfaces of a specified group will be shown. When executing a command without parameter, all physical interface current load will be displayed.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show interfaces utilization gigabitethernet 1/0/3-5,1/0/9
Port       Period, s     Sent,         Recv,         Frames Sent   Frames Recv
                         Kbit/s        Kbit/s
--------   -----------   -----------   -----------   -----------   -----------
gi1/0/3    5             0             0             0             0
gi1/0/4    5             0             0             0             0
gi1/0/5    5             0             0             0             0
gi1/0/9    5             0             0             0             0
```

shutdown

The command disables an interface being configured.

The use of a negative form (no) of the command enables the interface being configured.

Syntax

```
[no] shutdown
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-if-gi)# shutdown
```

Configured interface is disabled.

show system jumbo-frames

> ⚠ In the current firmware version this functionality is supported by ESR-200/1000/1200/1500/1511/1700 routers only

The command displays the jumbo frame function current status and status after device reboot.

Syntax

```
show system jumbo-frames
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show system jumbo-frames
Jumbo frames are disabled
Jumbo frames will be disabled after reset
```

speed

The command sets speed value for interface (interface group) being configured. The command can set the following modes: 10 Mbps, 100 Mbps, 1000 Mbps, 10 Gbps or auto.

The use of a negative form (no) of the command sets the default value.

Syntax

`speed <SPEED> <DUPLEX>`

`no speed`

Parameters

<SPEED> – speed value:

- 10M – 10Mbps speed;
- 100M – 100Mbps speed;
- 1000M – 1000Mbps speed;
- 10G – 10Gbps speed;
- auto – automatic mode selection (unavailable for 10G interfaces).

<DUPLEX> – transmitter receiver operation mode, takes the following values:

- full-duplex;
- half-duplex.

Default value

auto

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example 1

```
esr(config-if-te)# speed 10G
```

10 Gbps speed limit is set.

Example 2

```
esr(config-if-gi)# speed 10M full-duplex
```

10 Mbps speed limit is set, duplex.

switchport dot1q ethertype egress stag

The command configures EtherType for service VLAN in outgoing packets.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] switchport dot1q ethertype egress stag { 802.1q | 802.1ad }
```

Parameters

The command does not contain parameters.

Default value

802.1q (0x8100)

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example:

```
esr(config-if-gi)# switchport dot1q ethertype egress stag 802.1ad
```

switchport mode

The command specifies VLAN interface operation mode.

The use of a negative form (no) of the command sets the default value.

Syntax

`switchport mode <MODE>`

`no switchport mode`

Parameters

<MODE> – operation mode:

- access[1] – access interface, non-tagged interface for one VLAN;
- trunk[1] – interface that receives only tagged traffic excluding one VLAN that can be added by **switchport trunk native vlan** command described in Section switchport trunk native-vlan;
- general[2] – physical interface is switched to general mode;
- e1 – physical interface is switched to E1 mode (not applicable for aggregated interfaces).

Default value

access[1]

general[2]

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example:

```
esr(config-if-gi)# switchport mode trunk
```

[1] This command is supported on the ESR-10/12V/12VF/14VF/20/21/100/200/3100 routers only

[2] This command is supported on the ESR-1000/1200/1500/1511/1700 routers only

system jumbo-frames

The command enables Jumbo frame support. To bring the changes into effect, reboot the device.

The use of a negative form (no) of the command disables Jumbo frame support.

Syntax

```
[no] system jumbo-frames
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# system jumbo-frames
```

Physical interfaces

snmp trap link-status

The command enables sending snmp-trap about tunnel enabling/disabling.

The use of a negative form (no) of the command disables sending snmp-trap about tunnel enabling/disabling.

Syntax

```
[no] snmp trap link-status
```

Parameters

The command does not contain parameters.

Default value

Enabled.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

Example:

```
esr(config-if-gi)# snmp trap link-status
```

show interfaces protected-ports

> ⚠ In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

The command displays physical interfaces in group-based isolation mode.

Syntax

```
show interfaces protected-ports [ <IF> ]
```

Parameters

<IF> – a physical or aggregated interface's name specified in the form described in Section Types and naming order of router interfaces.

You may specify several interfaces in the command. If interface indexes are not specified, then the statuses of all interfaces of a specified group will be shown. If a certain interface is specified, the detailed information on this interface will be displayed. When executing a command without parameter, all physical interface statuses will be displayed.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show interfaces protected-ports
Interface    State           Community
---------    -------------   ---------
gi1/0/5      Protected       4
```

show interfaces sfp

The command displays SFP transceivers information.

Syntax

```
show interfaces sfp [ <IF> ]
```

Parameters

<IF> – a physical interface's name specified in the form described in Section Types and naming order of router interfaces. You may specify several interfaces in the command. If interface indexes are not specified, then the statuses of all interfaces of a specified group will be shown. If a certain interface is specified, the detailed information on this interface will be displayed. When executing a command without parameter, all system interface statuses will be displayed.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show interfaces sfp
Interface 'te1/0/1':
 SFP present:        Yes
 Connector Type:     LC
 Type:               SFP/SFP+
 Compliance code:    10G BASE-SR
 Laser wavelength:   850 nm
 Transfer distance: 300.00 m
 Vendor OUI:         24:00:00
 Vendor name:        Modultech
 Vendor PN:          MT-PP-85192-SR
 Vendor SN:          M1204011007
 Vendor date:        04.05.12
 Vendor revision:    1.0
 DDM supported:      Yes
 Temperature:        40.562 C
 Voltage:            3.3364 V
 Current:            6.004 mA
 RX Power:           0.0001 mW / -40.0000 dBm
 TX Power:           0.4398 mW / -3.5674 dBm
 RX LOS:             Yes
 TX Fault:           No
 TX Disable:         No
 Soft TX Disable:    No
Interface 'te1/0/2':
 SFP present:        Yes
 Connector Type:     SC
 Type:               SFP/SFP+
 Compliance code:    1000BASE-LX
 Laser wavelength:   1310 nm
 Transfer distance: 20.00 km
 Vendor OUI:         --
 Vendor name:        OEM
 Vendor PN:          APSB35123CXS20
 Vendor SN:          SG35224701333
 Vendor date:        12.12.12
 Vendor revision:    1.00
 DDM supported:      No
```

show interfaces switch-port configuration

> In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

The command displays physical interface configuration parameters.

Syntax

```
show interfaces switch-port configuration [ <IF> ]
```

Parameters

<IF> – a physical interface's name specified in the form described in Section Types and naming order of router interfaces.

You can specify several interfaces separated by commas ',' or you can specify the range of interfaces with '-'. If interface indexes are not specified, then the parameters of all interfaces of a specified group will be shown. When executing a command without parameter, all physical interface parameters will be displayed.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show interfaces switch-port configuration gigabitethernet 1/0/5-7
Port       Media     Duplex   Speed      Neg        Flow      Admin   Back
                                                    control   State   Pressure
--------   -------   ------   --------   --------   -------   -----   --------
gi1/0/5    none      Half     10 Mbps    Enabled    Off       Up      Disabled
gi1/0/6    none      Half     10 Mbps    Enabled    Off       Up      Disabled
gi1/0/7    none      Half     10 Mbps    Enabled    Off       Up      Disabled
```

show interfaces switch-port status

The command displays physical interfaces status.

Syntax

```
show interfaces switch-port status [ <IF> ]
```

Parameters

<IF> – a physical or aggregated interface's name specified in the form described in Section Types and naming order of router interfaces.

You can specify several interfaces separated by commas ',' or you can specify the range of interfaces with '-'. If interface indexes are not specified, then the statuses of all interfaces of a specified group will be shown. If a certain interface is specified, the detailed information on this interface will be displayed. When executing a command without parameter, all physical interface statuses will be displayed.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show interfaces switch-port status
Port       Media    Duplex   Speed   Neg        Flow   Link    Back       MDI       Port
                                                ctrl   State   Pressure   Mode      Mode
--------   ------   ------   -----   --------   ----   -----   --------   -------   -------
gi1/0/1    --       --       --      Enabled    --     Down    --         --        access
gi1/0/2    --       --       --      Enabled    --     Down    --         --        access
gi1/0/3    --       --       --      Enabled    --     Down    --         --        access
gi1/0/4    --       --       --      Enabled    --     Down    --         --        access
gi1/0/5    --       --       --      Enabled    --     Down    --         --        access
gi1/0/6    --       --       --      Enabled    --     Down    --         --        access

esr# show interfaces switch-port status gigabitethernet 1/0/2
Interface       gigabitethernet 1/0/2
  Status:       up
  Media:        copper
  Speed:        100 Mbps
  Duplex:       full
  Flow control: no
  MAC address:  a8:f9:b5:00:00:25
MAC status:
  Buffers full:            no
  Doing back pressure:     no
  Sending PAUSE frames:    no
  Receiving PAUSE frames:  no
  Auto-Negotiation done:   yes
  Sync fail:               no
```

threshold rx-utilization

The command specifies interface incoming load threshold to send snmp-trap eltexInterfaceRxUtilizationHigh and eltexInterfaceRxUtilizationHighOk.

The use of a negative form (no) of the command sets the default value.

Syntax

```
threshold rx-utilization <TH-HIGH> clear <TH-LOW>
```

```
no threshold rx-utilization
```

Parameters

<TH-HIGH> – threshold (in per cent) to send snmp-trap eltexInterfaceRxUtilizationHigh;

<TH-LOW> – threshold (in per cent) to send snmp-trap eltexInterfaceRxUtilizationHighOk.

Default value

<TH-HIGH> – 90%;

<TH-LOW> – 85%.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example:

```
esr(config-if-gi)# threshold rx-utilization 90 clear 80
```

threshold rx-utilization monitoring

The command enables sending snmp-trap eltexInterfaceRxUtilizationHigh and eltexInterfaceRxUtilizationHighOk.

The use of a negative form (no) of the command disables sending snmp-trap eltexInterfaceRxUtilizationHigh and eltexInterfaceRxUtilizationHighOk.

Syntax

```
[no] threshold rx-utilization monitoring
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example:

```
esr(config-if-gi)# threshold rx-utilization monitoring
```

threshold tx-utilization

The command sets interface upstream thresholds for sending snmp-traps eltexInterface T xUtilizationHigh and eltexInterface T xUtilizationHighOk.

The use of a negative form (no) of the command sets the default value.

Syntax

```
threshold tx-utilization <TH-HIGH> clear <TH-LOW>
```

```
no threshold tx-utilization
```

Parameters

<TH-HIGH> – threshold (in per cent) to send snmp-trap eltexInterfaceTxUtilizationHigh;

<TH-LOW> – threshold (in per cent) to send snmp-trap eltexInterfaceTxUtilizationHighOk.

Default value

<TH-HIGH> – 90%;

<TH-LOW> – 85%.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example:

```
esr(config-if-gi)# threshold tx-utilization 90 clear 80
```

threshold tx-utilization monitoring

The command is used to enable sending snmp-traps eltexInterface T xUtilizationHigh and eltexInterface T xUtilizationHighOk.

The use of the negative form of the command (no) disables sending snmp-traps eltexInterface T xUtilizationHigh and eltexInterface T xUtilizationHighOk.

Syntax

```
[no] threshold tx-utilization monitoring
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example:

```
esr(config-if-gi)# threshold tx-utilization monitoring
```

switchport community

> ⚠ In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

The command adds an interface to an isolation group. The command is relevant if the port is in group-based isolation mode.

The use of a negative form (no) of the command removes an interface from isolation group.

Syntax

```
switchport community <ID>
no switchport community
```

Parameters

<ID> – group identifier, takes values in the range of [1..30].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example:

```
esr(config-if-gi)# switchport community 10
```

switchport protected

> ⚠ In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

The command enables Private VLAN functionality and specifies an interface to which received packets can be sent.

The use of a negative form (no) of the command disables Private VLAN functionality.

Syntax

```
switchport protected <IF>
no switchport protected
```

Parameters

<IF> – an interface's name specified in the form described in Section Types and naming order of router interfaces.

Default value

Private VLAN functionality is disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example:

```
esr(config-if-gi)# switchport protected gigabitethernet 1/0/1
```

switchport protected-port

⚠ In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

The command switches an interface to a group-based isolation mode. According to this mode, traffic exchange is allowed for interfaces belonging to the same group, prohibited for interfaces from different groups and allowed for isolated and non-isolated interfaces.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] switchport protected-port
```

Parameters

None.

Default value

Interface not isolated.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example:

```
esr:esr(config-if-gi)# switchport protected-port
```

Aggregated interfaces

channel-group

The command adds a physical interface to channel aggregation group.

The use of a negative form (no) of the command removes an interface from channel aggregation group.

Syntax

```
channel-group <ID> mode <MODE>
no channel-group
```

Parameters

<ID> – sequence number of a channel aggregation group, takes values of [1..12].

<MODE> – mode of the channel aggregation group formation:

- auto – add interface to the dynamic aggregation group with the support of LACP;
- on – add interface to the static aggregation group.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example:

```
esr(config-if-gi)# channel-group 6 mode auto
```

lacp port-priority

The command sets LACP priority of Ethernet interface.

The use of a negative form (no) of the command sets the default value.

Syntax

`lacp port-priority <PRIORITY>`

`no lacp port-priority`

Parameters

<PRIORITY> – priority, set in the range of [1..65535].

Default value

1

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example:

```
esr(config-if-gi)# lacp port-priority 5000
```

lacp system-priority

The command sets the system priority for LACP.

The use of a negative form (no) of the command sets the default value.

Syntax

`lacp system-priority <PRIORITY>`

`no lacp system-priority`

Parameters

<PRIORITY> – priority, set in the range of [1..65535].

Default value

1

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# lacp system-priority 5000
```

lacp timeout

The command sets an administrative LACP timeout.

The use of a negative form (no) of the command sets the default value.

Syntax

`lacp timeout { short | long }`

`no lacp timeout`

Parameters

long – long timeout (90 seconds);

short – short timeout (3 seconds).

Default value

long

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example:

```
esr(config-if-gi)# lacp timeout short
```

port-channel load-balance

The command sets the load balancing mechanism for channel aggregation groups.

The use of a negative form (no) of the command sets the default value.

Syntax

```
port-channel load-balance {src-dst-mac-ip|src-dst-mac|src-dst-ip|src-dst-mac-ip-port}
no port-channel load-balance
```

Parameters for ESR-1000, ESR-1200 ESR-1500, ESR-1511 and ESR-1700

– src-dst-mac – balancing mechanism is based on source and destination MAC address;

– src-dst-ip – balancing mechanism is based on source and destination IP address;

– src-dst-mac-ip – balancing mechanism is based on source and destination MAC addresses and IP addresses;

src-dst-ip-port – balancing mechanism is based on source and destination IP addresses and tcp/udp ports;

– src-dst-mac-ip-port – balancing mechanism is based on source and destination MAC address, IP address and port.

Parameters for ESR-10, ESR-12V, ESR-12VF, ESR-14VF, ESR-20, ESR-21, ESR-100, ESR-200 and ESR-3100

active-backup – all traffic is sent to one interface and redirected to another if the first one has passed to down status;

src-dst-mac – balancing mechanism is based on the MAC addresses of a sender and receiver;

src-dst-mac-ip – balancing mechanism is based on source and destination MAC addresses and IP addresses;

src-dst-ip-port – balancing mechanism is based on source and destination IP addresses and tcp/udp ports;

Default value

src-dst-mac

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# port-channel load-balance src-dst-mac-ip
```

show interfaces port-channel

The command displays information on members of channel aggregation group.

Syntax

```
show interfaces port-channel [<ID>]
```

Parameters

<ID> – sequence number of a channel aggregation group, takes values of [1..12].

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show interfaces port-channel 1
load-balance: src-dst-mac
Channels   Ports
--------   ----------------------------------------
po1        gi1/0/21
```

show lacp counters

> ⚠ In the current firmware version this functionality is supported only by ESR-1000/1200/1500/1511/1700 routers

The command displays the statistics on LACP operation for Ethernet interface.

Syntax

```
show lacp counters [ <IF> ]
```

Parameters

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces.

You may specify several interfaces in the command. If interface indexes are not specified, then the statistics on all interfaces of a specified group will be shown. If a certain interface is specified, the detailed information on this interface will be displayed.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show lacp counters port-channel 2
Interface          Sent           Recv           Link failure
----------------   ------------   ------------   ------------
po2                42             814            2
```

show lacp interfaces

The command displays LACP information.

Syntax

```
show lacp interfaces [ <IF> ]
```

Parameters

<IF> – a physical or aggregated interface's name specified in the form described in Section Types and naming order of router interfaces.

You may specify several interfaces in the command. If interface indexes are not specified, then LACP information will be shown for all interfaces of a specified group.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show lacp interfaces port-channel 2
port-channel 2 [aggregator 1, active] ports count: 1
                   Actor Port             Partner Port
----------------   --------------------   --------------------
System Priority    32768                  1
System MAC         a8:f9:4b:aa:12:40      a8:f9:4b:83:01:80
Key                8000                   1
port-channel 2 [aggregator 2, backup] ports count: 1
                   Actor Port             Partner Port
----------------   --------------------   --------------------
System Priority    32768                  65535
System MAC         a8:f9:4b:aa:12:40      00:00:00:00:00:00
Key                8000                   FFFF
esr# show lacp interfaces gigabitethernet 1/0/1
gigabitethernet 1/0/1 [active] up
                   Actor Port     Partner Port
----------------   ------------   ------------
Port Priority      32768          1
LACP Activity      Active         Active
```

show lacp parameters

The command displays LACP configuration parameters for Ethernet interface.

Syntax

```
show lacp parameters [ <IF> ]
```

Parameters

<IF> – a physical interface's name specified in the form described in Section Types and naming order of router interfaces. The use of only physical interfaces is possible.

You may specify several interfaces in the command. If interface indexes are not specified, then the parameters of all interfaces of a specified group will be shown. If a certain interface is specified, the detailed information on this interface will be displayed.

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show lacp parameters tengigabitethernet 1/0/2
   LACP parameters
   ~~~~~~~~~~~~~~~
Interface   Port Priority   Timeout   Mode
---------   -------------   -------   -------
te1/0/2    32768           Short     Active
```

E1/multilink

enable

The command enables a PPP user.

The use of a negative form (no) of the command disables a PPP user.

Syntax

`[no] enable`

Parameters

The command does not contain parameters.

Default value

PPP user disabled.

Required privilege level

10

Command mode

CONFIG-PPP-USER

Example:

```
esr(config-ppp-user)# enable
```

ip tcp header-compression

The command enables TCP header compression protocol. The protocol is used to improve performance of low-speed channels.

The use of a negative (no) form of the command disables TCP header compression protocol.

Syntax

```
[no] ip tcp header-compression
```

Parameters

None.

Required privilege level

10

Command mode

CONFIG-E1

CONFIG-MULTILINK

Example:

```
(config-e1)# ip tcp header-compression
```

ip tcp compression-connections

The command specifies a number of simultaneous tcp connections for which TCP header compression protocol will be used.

The use of a negative (no) form of a command sets a default number of simultaneous tcp connections for which TCP header compression protocol will be used.

Syntax

```
ip tcp compression-connections <NUMBER>
no ip tcp compression-connections
```

Parameters

<NUMBER> – a number of simultaneous tcp connections for which a tcp headers compression protocol will be used. Can take values [2..16].

Default value

16

Required privilege level

10

Command mode

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr:esr(config-e1)# ip tcp compression-connections 32
esr:esr(config-e1)#
```

mrru

The command specifies the maximum size of a received packet for MLPPP interface.

The use of a negative form (no) of the command sets the default mrru value.

Syntax

`mrru { <MRRU> }`

`no mrru`

Parameters

<MRRU> – maximum size of a received packet for MLPP interface, takes value in the range of [1500..10000].

Default value

1500

Required privilege level

10

Command mode

CONFIG-MULTILINK

Example:

```
esr(config-multilink)# mrru 1700
```

password

Set encrypted or unencrypted password for a specific user to authenticate the remote party. User password is kept in configuration in encrypted form. During configuration you can specify unencrypted password or copy encrypted password from another device.

The use of a negative form (no) of the command removes a user's password.

Syntax

`password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }`

`no password`

Parameters

<CLEAR-TEXT> – unencrypted password, set by the string of [1..64] characters, may include [0-9a-fA-F] characters;

<ENCRYPTED-TEXT> – unencrypted password, set by the string of [2..128] characters.

> ⚠ The password are kept in encrypted form independently of format used when entering the command.

Required privilege level

15

Command mode

CONFIG-PPP-USER

Example:

```
esr(config-ppp-user)# password ascii-text 01234567
```

ppp authentication chap

The command enables CHAP authentication.

The use of a negative form (no) of the command disables the authentication.

Syntax

```
[no] ppp authentication chap
```

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

CONFIG-SERIAL

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-e1)# ppp authentication chap
```

ppp chap hostname

Specify the router name that is sent to a remote party for CHAP authentication.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ppp chap hostname <NAME>
no ppp chap hostname
```

Parameters

<NAME> – router name, set by the string of up to 31 characters.

Default value

Device system name (hostname).

Required privilege level

15

Command mode

CONFIG-SERIAL

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-e1)# ppp chap hostname esr1
```

ppp chap password

Specify the password that is sent with the router name to a remote party for CHAP authentication.

The use of a negative form (no) of the command removes the password.

Syntax

```
ppp chap password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no ppp chap password
```

Parameters

<CLEAR-TEXT> – unencrypted password, set by the string of [1..64] characters, may include [0-9a-fA-F] characters;

<ENCRYPTED-TEXT> – unencrypted password, set by the string of [2..128] characters.

⚠ The password is kept in encrypted form independently of format used when entering the command.

Required privilege level

15

Command mode

CONFIG-SERIAL

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-e1)# ppp chap password ascii-text 01234567
```

ppp chap refuse

The command enables authentication ignoring.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] ppp chap refuse
```

Parameters

The command does not contain parameters.

Default value

Authentication ignoring is disabled.

Required privilege level

15

Command mode

CONFIG-SERIAL

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-e1)# ppp chap refuse
```

ppp chap username

The command specifies a user for remote party authentication and switches to user configuration mode.

The use of a negative form (no) of the command removes a specified user.

Syntax

```
[no] ppp chap username <NAME>
```

Parameters

<NAME> – user name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-SERIAL

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-e1)# ppp chap username xap
```

ppp ipcp accept-address

The command ensures any non-null IP address to be accepted as a local IP address from the neighbor.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] ppp ipcp accept-address
```

Parameters

The command does not contain parameters.

Default value

IP address reception is prohibited.

Required privilege level

10

Command mode

CONFIG-SERIAL

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-e1)# ppp ipcp accept-address
```

ppp ipcp remote-address

The command sets IP address that is sent to a remote party for the further allocation.

The use of a negative form (no) of the command removes remote party IP address.

Syntax

```
ppp ipcp remote-address <ADDR>
no ppp ipcp remote-address
```

Parameters

<ADDR> – IP address of a remote gateway.

Required privilege level

10

Command mode

CONFIG-SERIAL

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-e1)# ppp ipcp remote-address 192.168.1.2
```

ppp max-configure

The command sets the amount of attempts to send Configure-Request packets before the remote peer is found to be unable to respond.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ppp max-configure <VALUE>
no ppp max-configure
```

Parameters

<VALUE> – time in seconds, takes values of [1..255].

Default value

10

Required privilege level

10

Command mode

CONFIG-SERIAL

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-e1)#i ppp max-configure 4
```

ppp max-failure

The command sets the amount of attempts to send Configure-NAK packets before all options are confirmed.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ppp max-failure <VALUE>
no ppp max-failure
```

Parameters

<VALUE> – time in seconds, takes values of [1..255].

Default value

10

Required privilege level

10

Command mode

CONFIG-SERIAL

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-if-gi)#i ppp max-failure 3
```

ppp max-terminate

The command sets the amount of attempts to send Terminate-Request packets before the session is aborted.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ppp max-terminate <VALUE>
no ppp max-terminate
```

Parameters

<VALUE> – time in seconds, takes values of [1..255].

Default value

2

Required privilege level

10

Command mode

CONFIG-SERIAL

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-if-gi)#i ppp max-terminate 4
```

ppp mru

The command sets MRU (Maximum Receive Unit) size for the interface.

The use of a negative form (no) of the command sets the default MRU value.

Syntax

```
ppp mru <MRU>
no ppp mru
```

Parameters

<MRU> – MRU value, takes values in the range of [552..1500].

Default value

1500

Required privilege level

10

Command mode

CONFIG-SERIAL

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-if-gi)# mru 1400
```

ppp multilink

The command enables MLPPP mode on E1 interface.

The use of a negative form (no) of the command disables MLPPP mode.

Syntax

```
[no] ppp multilink
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-E1

Example:

```
esr(config-e1)# ppp multilink
```

ppp multilink-group

The command adds E1 interface to aggregation group.

The use of a negative form (no) of the command removes an interface from aggregation group.

Syntax

`ppp multilink-group <GROUP-ID>`

Parameters

<GROUP-ID> – group identifier, takes values in the range of [1..4].

Required privilege level

10

Command mode

CONFIG-E1

Example:

```
esr(config-e1)# ppp multilink-group 1
```

ppp timeout keepalive

The command specifies the time interval in seconds after which the router sends a keepalive message.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ppp timeout keepalive [ <TIME >]
no ppp timeout keepalive
```

Parameters

<TIME> – time in seconds, takes values of [1..32767].

Default value

10

Required privilege level

10

Command mode

CONFIG-SERIAL

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-if-gi)# ppp timeout keepalive 200
```

ppp timeout retry

The command specifies the time interval in seconds after which the router reiterates the request to establish a session.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ppp timeout retry <TIME>
no ppp timeout retry
```

Parameters

<TIME> – time in seconds, takes values of [1..255].

Default value

3

Required privilege level

10

Command mode

CONFIG-SERIAL

CONFIG-E1

CONFIG-MULTILINK

Example:

```
esr(config-if-gi)# ppp timeout retry 3
```

show controllers e1

The command displays information on E1 controllers.

Syntax

```
show controllers e1 [<IF>]
```

Parameters

<IF> – a physical interface's name specified in the form described in Section Types and naming order of router interfaces;

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show controllers e1
Interface 'te1/0/1':
 SFP present:     Yes
 SFP Vendor name: NSC-COM
 SFP Vendor PN:   611.900
 Line code:       HDB3
 Clock source:    Internal
 Timeslot:        24
 Invert Data:     No
 Framing CRC4:    No
 Loopback:        --
 CRC algorithm:   FCS16
 E1 Link:         Down
 E1 Synced:       No
 E1 RX AIS:       No
 E1 RX RAI:       No
```

switchport e1 slot

The command binds e1 port to a physical interface. The use of a negative form (no) of the command switches to standard mode.

Syntax

```
[no] switchport e1 slot <SLOT>
```

Parameters

<SLOT> – slot identifier, takes values in the range of [0..12].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example:

```
esr(config-if-gi)# switchport e1 0
```

switchport e1 clock source

The command defines a e1 interface's synchronization source.

The use of a negative form (no) of the command sets the default value.

Syntax

```
switchport e1 clock source { internal | line }
no switchport e1 clock source
```

Parameters

internal – internal synchronization source is used;

line – a signal from the line is used for synchronization.

Default value

internal

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example:

```
esr:esr(config-if-gi)# switchport e1 clock source line
```

switchport e1 crc

The command defines a mode for integrity check of data transmitted.

The use of a negative form (no) of the command sets the default value.

Syntax

```
switchport e1 crc { 16 | 32 }
no switchport e1 crc
```

Parameters

16 – a 16-bit Cyclic redundancy check (CRC) algorithm is used;

32 – a 32-bit Cyclic redundancy check (CRC) algorithm is used;

Default value

16

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example:

```
esr:esr(config-if-gi)# switchport e1 crc 16
```

switchport e1 framing

The command enables integrity check based on CRC4 algorithm

The use of a negative form (no) of the command sets the default value.

Syntax

```
switchport e1 framing { crc4 | no-crc4 }
no switchport e1 framing
```

Parameters

crc4 – enable CRC4-based integrity check;

no-crc4 – disable CRC4-based integrity check.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example:

```
esr:esr(config-if-gi)# switchport e1 framing crc4
```

switchport e1 invert data

The command enables reverse data sending.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] switchport e1 invert data
```

Parameters

None.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example:

```
esr:esr(config-if-gi)# switchport e1 invert data
```

switchport e1 linecode

The command defines an algorithm for data encryption to transmit it via a physical channel.

The use of a negative form (no) of the command sets the default value.

Syntax

```
switchport e1 linecode { ami | hdb3 }
```

Parameters

ami – use ami algorithm;

hdb3 – use hdb3 algorithm.

Default value

hdb3

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example:

```
esr:esr(config-if-gi)# switchport e1 linecode ami
```

switchport e1 timeslots

The command defines a number of 64kbps channels in one E1.

The use of a negative form (no) of the command sets the default value.

Syntax

```
switchport e1 timeslots < RANGE >
no switchport e1 timeslots
```

Parameters

< RANGE > – the number of 64 kbps channels, takes values [1..31].

Default value

31

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example:

```
esr:esr(config-if-gi)# switchport e1 timeslots 16
```

switchport e1 unframed

The command enables the mode according to which E1 is not splitted to 64kbps channels and used as a single one.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] switchport e1 unframed
```

Parameters

None.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example:

```
esr:esr(config-if-gi)# switchport e1 unframed
```

switchport mode e1

The command switches a physical port to SFP e1 module operation mode.

The use of a negative form (no) of the command switches a port to standard mode.

Syntax

```
[no] switchport mode e1
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example:

```
esr(config-if-gi)# switchport mode e1
```

Serial interfaces

> ⚠ Only for ESR-21

chat-script

The command is used to create chat-script, which can be used when connecting via dialup.

The use of a negative form (no) of the command removes the chat-script.

Syntax

```
chat-script <NAME> <TEXT>
no chat-script { <NAME> | all }
```

Parameters

<NAME> – script name, set by the string of up to 31 characters.

<TEXT> – script content, set by the string of up to 255 characters. It is necessary to enclose the script in parentheses and shield spaces and specialized symbols with the "\" symbol.

all — the key used to delete all created scripts.

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# chat-script DIAL "\(ABORT ERROR ABORT BUSY \"\" \"ATZ\" OK \"ATDTT\" TIMEOUT 30
CONNECT\)"
```

clear line aux

This command is used to remove serial interface sessions.

Syntax

```
clear line aux [ <NUM> ]
```

Parameters

<NUM> – a number of a serial interface in the range [1..3].

Required privilege level

15

Command mode

ROOT

Example:

```
esr# clear line aux 2
```

databits

The command specifies a number of data bits sent.

The use of a negative form (no) of the command sets the default value.

Syntax

```
databits <BITS>
no databits
```

Parameters

<BITS> – a number of data bits sent. Takes values [7..8]

Default value

8

Required privilege level

15

Command mode

CONFIG-LINE-AUX

Example:

```
esr(config-line-aux)# databits 7
```

dialer

This command is used to enable dialer function.

The use of a negative form (no) of the command disables this feature.

Syntax

`[no] dialer`

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-SERIAL

Example:

```
esr(config-serial)# dialer
```

dialer idle-timeout

This command sets the time of connection holding in the absence of traffic.

The use of a negative form (no) of the command sets the default value.

Syntax

```
dialer idle-timeout <TIME>
no dialer idle-timeout
```

Parameters

<TIME> – time of connection holding in the absence of traffic in seconds, takes value in the range [1..65535].

Default value

30

Required privilege level

10

Command mode

CONFIG-SERIAL

Example:

```
esr(config-serial)# dialer idle-timeout 210
```

dialer in-band

This command enables the V.25bis mode at the serial interface.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] dialer in-band
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-SERIAL

Example:

```
esr(config-serial)# dialer in-band
```

dialer map

This command specifies the correspondence of IP subnetworks and dial-up numbers for access to these subnetworks.

The use of a negative form (no) of the command sets the default value.

Syntax

```
dialer map ip <SUBNET> name <PPPUSER> [ modem-script <SCRIPT-NAME> ] <PHONE-NUM>
no dialer map ip
```

Parameters

<SUBNET> – destination address, specified as AAA.BBB.CCC.DDD/NN, where AAA-DDD may take values [0..255] and NN may take values [1..32].

<PPPUSER> – user name for dialer, set by the string of up to 31 characters.

<SCRIPT-NAME> – name of the dialer script, set by the string of up to 31 characters.

<PHONE-NUM> – number for dialer. Specified by a string of up to 15 characters, it is possible to use only digits.

Default value

None

Required privilege level

10

Command mode

CONFIG-SERIAL

Example:

```
esr(config-serial)# dialer map ip 192.168.33.0/27 name SITE12 5512
```

dialer string

This command sets the phone number for dialup modem connection.

The use of the negative form of the command (no) removes the phone number for connection establishing via dialup modem.

Syntax

```
dialer string <PHONE-NUM>
no dialer string
```

Parameters

<PHONE-NUM> – phone number for dialer. Specified by a string of up to 15 characters, it is possible to use only digits.

Default value

None

Required privilege level

15

Command mode

CONFIG-SERIAL

Example:

```
esr(config-serial)# dialer string 3835401
```

flowcontrol

The command sets a mode of data flow control.

The use of a negative form (no) of the command sets the default value.

Syntax

```
flowcontrol <MODE>

no flowcontrol
```

Parameters

<MODE> – data flow control mode. Takes the following values:

- software – sowtware flow control;
- hardware – hardware flow control;
- disabled – flow control disabled.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-LINE-AUX

Example:

```
esr(config-line-aux)# flowcontrol software
```

line aux

The command switches to the configuration mode of a serial interface.

Syntax

`line aux <NUM>`

Parameters

<NUM> – a number of a serial interface from the range [1..3].

Required privilege level

15

Command mode

CONFIG

Example:

```
esr(config)# line aux 1
esr(config-line-aux)#
```

modem inout

The command switches a serial interface to the mode of operation with a modem.

The use of a negative form of the command (no) switches a serial interface to the console server mode.

Syntax

`modem inout`

`no modem inout`

Parameters

None.

Default value

By default, a serial interface operates in the console server mode.

Required privilege level

15

Command mode

CONFIG-LINE-AUX

Example:

```
esr(config-line-aux)# modem inout
```

parity

The command enables a parity bit setting mode.

The use of a negative form (no) of the command sets the default value.

Syntax

```
parity <MODE>
no parity
```

Parameters

<MODE> – parity bit setting mode. Takes the following values:

- odd – a check for oddness
- even –a check for evenness
- none – parity bit is not set.

Default value

none.

Required privilege level

15

Command mode

CONFIG-LINE-AUX

Example:

```
esr(config-line-aux)# parity even
```

show line aux

The command is used to display serial interfaces status.

Syntax

`show line aux [ <NUM> ]`

Parameters

<NUM> – a number of a serial interface in the range [1..3].

Required privilege level

10

Command mode

ROOT

Example:

```
esr# show line aux
AUX Line 1
 Baud rate:              115200
 Databits:               8
 Parity:                 NONE
 Stopbits:               1
 Flowcontrol:            Disabled
 Timeout Exec (min):     300
 Telnet port:            2001
 Modem mode:             Disabled
```

speed

The command sets a speed for a serial interface.

The use of a negative form (no) of the command sets the default value.

Syntax

`speed <SPEED>`

`no speed`

Parameters

<SPEED> – a speed of a serial interface in bps. Takes the following values:

- 300;
- 1200;
- 2400;
- 4800;
- 9600;
- 19200;
- 38400;
- 57600;
- 115200.

Default value

115200.

Required privilege level

15

Command mode

CONFIG-LINE-AUX

Example:

```
esr(config-line-aux)# speed 9600
```

stopbits

The command sets the number of stop bits transmitted.

The use of a negative form (no) of the command sets the default value.

Syntax

```
stopbits <STOP-BITS>
no stopbits
```

Parameters

<STOP-BITS> – the number of stop bits transmitted. Takes values in the range of [1..2].

Default value

1

Required privilege level

15

Command mode

CONFIG-LINE-AUX

Example:

```
esr(config-line-aux)# stopbits 2
```

transport telnet port

In the command a TCP port number for console server mode is specified. When connecting to the router's IP address and to a TCP port configured by the command via telnet, a user will be connected to a console interface of a device connected to the router's serial interface.

The use of a negative (no) form of a command deletes a TCP port number for console server mode.

Syntax

```
transport telnet port <PORT>
no transport telnet port
```

Parameters

<PORT> – TCP port number for console server mode. Takes values in the range of [1..65535].

Default value

None

Required privilege level

15

Command mode

CONFIG-LINE-AUX

Example:

```
esr(config-line-aux)# transport telnet port 2001
```

Wireless modems

allowed-auth

The command specifies method of user authentication in a mobile network.

The use of a negative form (no) of the command sets the default authentication method.

Syntax

```
allowed-auth <TYPE>
no allowed-auth
```

Parameters

<TYPE> – method of user authentication in a mobile network [none, PAP, CHAP, MSCHAP, MSCHAPv2, EAP]

Default value

PAP

Required privilege level

15

Command mode

CONFIG-CELLULAR-PROFILE

Example:

```
esr(config-cellular-profile)# allowed-auth MSCHAP
```

allowed-mode

The command allows the use of the mode while USB-modem operation.

The use of a negative form (no) of the command removes a permission to use the mode.

Syntax

```
[no] allowed-mode <MODE>
```

Parameters

<MODE> – acceptable USB modem operation mode [2g, 3g, 4g]

Default value

The modes displayed as 'Allowed modes:' in show cellular status modem <ID> command output are allowed.

Required privilege level

10

Command mode

CONFIG-CELLULAR-MODEM

Example:

```
esr(config-cellular-modem)# allowed-mode 4g
```

apn

The command specifies mobile network access point.

Syntax

`apn <NAME>`

Parameters

<NAME> – mobile network access point, set by the string of up to 255 characters.

Default value

Not configured.

Required privilege level

10

Command mode

CONFIG-CELLULAR-PROFILE

Example:

```
esr(config-cellular-profile)# apn internet
```

cellular modem

The command creates USB modem with a certain identifier and switches to USB modem configuration mode.

The use of a negative form (no) of the command removes a configured USB modem.

Syntax

`[no] cellular modem <ID>`

Parameters

<ID> – USB modem identifier, set in the range of [1..10].

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# cellular modem 1
```

cellular profile

The command creates parameter profile for USB modem with a certain identifier and switches to profile configuration mode.

The use of a negative form (no) of the command removes a configured parameter profile of USB modem.

Syntax

```
[no] cellular profile <ID>
```

Parameters

<ID> – identifier of USB modem parameter profile, set in the range of [1..10].

Required privilege level

10

Command mode

CONFIG

Example:

```
esr(config)# cellular profile 1
```

device

The command specifies the identifier of connected modem’s USB port.

The use of a negative form (no) of the command removes a set identifier.

Syntax

```
device <WORD>
no device
```

Parameters

<WORD> – identifier of connected modem’s USB port, set in the range of [1..12].

The identifier should match the identifier in 'USB port device' field in show cellular status modem command output.

Default value

Not specified.

Required privilege level

10

Command mode

CONFIG-CELLULAR-MODEM

Example:

```
esr(config-cellular-modem)# device 1-1
```

enable

The command enables USB modem.

The use of a negative form (no) of the command disables USB modem.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

USB modem is disabled.

Required privilege level

10

Command mode

CONFIG-CELLULAR-MODEM

Example:

```
esr(config-cellular-modem)#
```

ip-version

The command restricts the ability to use IP addresses families in a mobile network.

The use of a negative form (no) of the command allows the use of both IPv4 and IPv6 families addressing.

Syntax

```
ip-version { ipv4 | ipv6 }
no ip-version
```

Parameters

ipv4 – ipv4 family;

ipv6 – ipv6 family;

Default value

Both IPv4 and IPv6 addressing is allowed.

Required privilege level

10

Command mode

CONFIG-CELLULAR-PROFILE

Example:

```
esr(config-cellular-profile)# ip-version ipv4
```

mru

The command sets the maximum size of a received packet.

The use of a negative form (no) of the command sets the default value.

Syntax

```
mru { <MRU> }
no mru
```

Parameters

<MRU> – MRU value, takes values in the range of [128..16383].

Default value

1500

Required privilege level

10

Command mode

CONFIG-CELLULAR-MODEM

Example:

```
esr(config-cellular-modem)# mru 1476
```

number

The command sets a dial-up number for connection to a mobile network.

The use of a negative form (no) of the command removes a dial-up number for connection to a mobile network.

Syntax

`number <WORD>`

`no number`

Parameters

<WORD> – dial-up number for connection to a mobile network, set by the string from 1 to 15 characters.

Default value

The number is not specified.

Required privilege level

10

Command mode

CONFIG-CELLULAR-PROFILE

Example:

```
esr(config-cellular-profile)# number *99#
```

password

The command sets the password of mobile network user in unencrypted or encrypted form. User password is kept in configuration in encrypted form. During configuration you can specify unencrypted password or copy encrypted password from another device.

The use of a negative form (no) of the command removes a user's password.

Syntax

```
password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no password
```

Parameters

<CLEAR-TEXT> – unencrypted password, set by the string of [1..64] characters, may include [0-9a-fA-F] characters;

<ENCRYPTED-TEXT> – unencrypted password, set by the string of [2..128] characters.

> ⚠ The password are kept in encrypted form independently of format used when entering the command.

Required privilege level

15

Command mode

CONFIG-PPP-USER

Example:

```
esr(config-ppp-user)# password ascii-text 01234567
```

pin

The command sets SIM card unblock code.

The use of a negative form (no) of the command removes SIM card unblock code.

Syntax

```
pin <WORD>
no pin
```

Parameters

<WORD> – SIM card unblock code [4..8] characters. Only digits are allowed.

Default value

PIN is not specified.

Required privilege level

15

Command mode

CONFIG-CELLULAR-MODEM

Example:

```
esr(config-cellular-modem)# pin 4856
```

preferred-mode

The command sets a preferable operation mode of USB modem in a mobile network.

The use of a negative form (no) of the command removes a preferable operation mode of USB modem in a mobile network.

Syntax

```
preferred-mode { <MODE> }
no preferred-mode
```

Parameters

<MODE> – preferable USB modem operation mode [2g, 3g, 4g].

Default value

Defined by USB modem.

Required privilege level

10

Command mode

CONFIG-CELLULAR-MODEM

Example:

```
esr(config-cellular-modem)# preferred-mode 4g
```

profile

The command assigns a created parameter profile for USB modem.

The use of a negative form (no) of the command removes a created parameter profile for USB modem.

Syntax

```
profile <ID>
```

```
no profile
```

Parameters

<ID> – identifier of USB modem parameter profile, set in the range of [1..10].

Default value

Parameter profile ID is not specified.

Required privilege level

10

Command mode

CONFIG-CELLULAR-MODEM

Example:

```
esr(config-cellular-modem)# profile 1
```

show cellular configuration modem

The command displays the information on configured USB modems

Syntax

```
show cellular configuration modem [ <ID> ]
```

Parameters

<ID> – USB modem identifier, set in the range of [1..10].

When USB modem number is not specified, a table with brief information on all configured USB modems is displayed.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# show cellular configuration modem
Number   State      Description                       USB port device    Profile
modem
------   --------   ------------------------------    ---------------    --------
1        Enabled    megafon                           1-1                1
2        Enabled    mts                               1-2                2

esr# show cellular configuration modem 2
State:             Enabled
Description:       mts
USB port device:   1-2
Pin:               --
MRU:               --
MTU:               1500
Preferred mode:    none
Allowed modes:     all
Profile:           2
Description:          MTS
  User name:            mts
  Number:               *99#
  APN:                  internet.mts.ru
  Password (encrypted): 91A010
  IP version:           both
  Allowed auth:         EAP
Security zone:     --
```

show cellular configuration profile

The command displays the information on configured USB modems profiles.

Syntax

```
show cellular configuration profile [ <ID> ]
```

Parameters

<ID> – USB modem identifier, set in the range of [1..10].

When USB modem number is not specified, a table with brief information on all configured USB modems is displayed.

Required privilege level

10

Command mode

ROOT

Example:

```
esr# show cellular configuration profile
Number    User name          APN                Number             Description
profile
-------   ----------------   ----------------   ----------------
------------------------------
1         gdata              internet           *99#               MEGAFON
2         mts                internet.mts.ru    *99#               MTS

esr# show cellular configuration profile 1
Description:          MEGAFON
User name:            gdata
Number:               *99#
APN:                  internet
Password (encrypted): 9BB00279B1
IP version:           both
Allowed auth:         EAP
```

show cellulas status modem

The command displays information on connected USB modems' status.

Syntax

`show cellular status modem [ <ID> ]`

Parameters

<ID> – USB modem identifier, set in the range of [1..10].

Without specifying a USB modem number, a table with brief information on all detected USB modems is displayed

Required privilege level

1

Command mode

ROOT

Example:

```
esr# show cellular status modem
Number   USB port   Manufacturer   Model        Current state   Inteface         Link
device                                                                           state
------   --------   ------------   ----------   -------------   ---------------   -----
1        1-1        huawei         E3372        disabled        --                Down

esr# show cellulars status modem 1
Interface 'modem 1' status information:
    USB port device:         1-1
    Manufacturer:            huawei
    Model:                   E3372
    Revision:                21.180.01.00.00
    IMEI:                    861821036192893
    Status SIM lock:         --
    Status unlock retries:   sim-pin (3) sim-pin2 (3) sim-puk (10) sim-puk2 (10)
    Current state:           disabled
    Access tech:             unknown
    Signal level:            0
    Support modes:
        allowed 2G; preferred none;
        allowed 3G; preferred none;
        allowed 4G; preferred none;
        allowed 2G 3G 4G; preferred none;
    Allowed modes:           2G 3G 4G
    Preferred modes:         none
    Type IP:                 IPv4
    Operator name:
    Registration:            unknown
```

user

The command specifies mobile network user name. After executing this command, the router enters the user parameters configuration mode.

The use of a negative form (no) of the command removes mobile network user name.

Syntax

```
[no] user <NAME>
```

Parameters

<NAME> – user name, set by the string from 1 to 31 characters.

Default value

User is not created.

Required privilege level

15

Command mode

CONFIG-CELLULAR-PROFILE

Example:

```
esr(config-cellular-profile)# user gdata
```

Phone port configuration

> ⚠ This functionality is supported only by ESR-12V/12VF/14VF routers

authentication name

The command assigns a user name that is used for authentication on SIP server (and registration server).

The use of a negative form (no) of the command removes the value.

Syntax

```
authentication name {<LOGIN> | as-phone}
no authentication name
```

Parameters

<LOGIN> – user name that is used for authentication, set by the string of up to 31 characters or by as-phone directive which means that user name for authentication will be equal to the phone number;

As-phone – user name for authentication, equal to the phone number.

Default value

None

Required privilege level

15

Command mode

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxo)# authentication name userlogin
```

authentication password

The command assigns a password that is used for authentication on SIP server (and registration server).

The use of a negative form (no) of the command removes the value.

Syntax

```
authentication password { <PASS> | encrypted <ENCRYPTED-PASS>}
no authentication password
```

Parameters

<PASS> – authentication password, set by the string of up to 16 characters.

<ENCRYPTED-PASS> – hash password via sha512 algorithm, set by the string of [2..32] characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# authentication password superpassword
```

call-forwarding busy

The command allows the use of CFB (Call Forward at Busy) service – call forwarding to a specified number at busy subscriber.

The use of a negative form (no) of the command restricts the use of *CFB* service..

Syntax

```
[no] call-forwarding busy
```

Parameters

The command does not contain parameters.

Default value

CFB service is disabled.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# call-forwarding busy
```

call-forwarding busy number

The command assigns a number, which incoming calls are forwarded to, with *CFB* service enabled.

The use of a negative form (no) of the command removes a call-forwarding busy number.

Syntax

```
call-forwarding busy number <PHONE>
no call-forwarding busy number
```

Parameters

<PHONE> – a number to which incoming calls are forwarded at busy subscriber, set by the string of up to 50 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# call-forwarding busy number 4596
```

call-forwarding no-answer

The command allows the use of *CFNA (Call Forward at No Answer)* service – call forwarding to a specified number at no subscriber answer.

The use of a negative form (no) of the command restricts the use of *CFNA* service.

Syntax

```
[no] call-forwarding no-answer
```

Parameters

The command does not contain parameters.

Default value

CFNA service is disabled.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# call-forwarding no-answer
```

call-forwarding no-answer number

The command assigns a number, which incoming calls are forwarded to, with *CFNA* service enabled.

The use of a negative form (no) of the command removes a call-forwarding no-answer number.

Syntax

```
call-forwarding no-answer number <PHONE>
no call-forwarding no-answer number
```

Parameters

<PHONE> – number to which incoming calls are forwarded at no subscriber answer, set by the string of up to 50 characters

Default value

None

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# call-forwarding no-answer number 4685
```

call-forwarding no-answer timeout

The command specifies time interval, within which call forwarding is performed at subscriber no answer, with *CFNA* service enabled.

The use of a negative form (no) of the command sets the default value.

Syntax

```
call-forwarding no-answer timeout <TIME>
no call-forwarding no-answer timeout
```

Parameters

<TIME> – time interval in seconds within which call forwarding is performed at subscriber no answer, takes the values of [0-120].

Default value

0

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# call-forwarding no-answer timeout 15
```

call-forwarding unconditional

The command enables the use of CFU (Call Forward Unconditional) service – all incoming calls are forwarded unconditionally to a specified number.

The use of a negative form (no) of the command restricts the use of *CFB* service..

Syntax

```
[no] call-forwarding unconditional
```

Parameters

The command does not contain parameters.

Default value

CFU service is disabled.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# call-forwarding unconditional
```

call-forwarding unconditional number

The command specifies a call-forwarding unconditional number.

The use of a negative form (no) of the command removes a call-forwarding unconditional number.

Syntax

```
call-forwarding unconditional number <PHONE>
no call-forwarding unconditional number
```

Parameters

<PHONE> – number to which incoming calls are forwarded with CFU service enabled, set by the string of up to 50 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# call-forwarding unconditional number 4685
```

call-waiting

The command enables the use of *'Call Waiting'* service.

The use of a negative form (no) of the command restricts the use of *'Call Waiting'* service.

Syntax

`[no] call-waiting`

Parameters

The command does not contain parameters.

Default value

CW service is disabled.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

CONFIG-VOIP-PROFILE

Example:

```
esr(config-voice-port-fxs)# call-waiting
```

caller-id mode

The command selects Caller ID distribution mode.

The use of a negative form (no) of the command disables Caller ID distribution.

Syntax

```
caller-id mode <MODE>
no caller-id mode
```

Parameters

<MODE> – Caller ID distribution mode, may take the following values:

- auto – automatic detection of Caller-id method being used. Available only on FXO port;
- dtmf – Caller ID detection by DTMF method. The number is served between the first and second ringing tones by double frequency DTMF ringings;
- fsk-bell – Caller ID and name detection by FSK method according to Bell202 standard. The number is served between the first and second ringing tones by a stream of data with a frequency modulation;
- fsk-v23 – Caller ID and name detection by FSK method according to ITU-T V.23 standard. Issuance of the number in a line is carried out between the first and the second ringing signal by data stream with frequency modulation.

Default value

Disables Caller ID.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

CONFIG-VOIP-PROFILE

Example:

```
esr(config-voice-port-fxs)# caller-id mode fsk-v23
```

dial-type

The command selects the mode of dialing by FXO set.

The use of a negative form (no) of the command restores the default value.

Syntax

```
dial-type <MODE>
```

```
no dial-type
```

Parameters

<MODE> – mode of dialing by FXO set, may take the following values:

- dtmf – dialing by DTMF signal transmission;
- pulse – dialing by sequential interlocking and unlocking of a phone line.

Default value

dtmf

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# dial-type pulse
```

enable

The command enables a profile or a rule set.

The use of a negative form (no) of the command disables a profile or a set of rules.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

Not enabled.

Required privilege level

10

Command mode

CONFIG-VOIP-PROFILE

Example:

```
esr(config-config-voip-sip-proxy)# enable
```

flash call-transfer

The command selects flash function (flash call transfer) usage mode.

The use of a negative form (no) of the command returns the default value.

Syntax

```
flash call-transfer <METHOD>
```

```
no flash call-transfer
```

Parameters

<METHOD> – flash function usage mode, may take the following values:

- attended – flash is proceeded by the device locally (call transfer is carried out after the connection with the third subscriber is established);
- unattended – flash is proceeded by the device locally (call transfer is carried out after the third subscriber dial ends);
- transmit-flash – flash transfer to channel (by any method configured with transfer flash command in SIP profile parameters);
- local-transfer – call transfer within the device, without sending REFER message.

Default value

transmit-flash

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

CONFIG-VOIP-PROFILE

Example:

```
esr(config-voice-port-fxs)# flash call-transfer attended
```

flash call-transfer mode

The command selects call transfer mode – configuration is available only for Attended calltransfer and Local calltransfer and is in charge for call transfer service enable mode.

The use of a negative form (no) of the command returns the default value.

Syntax

```
flash call-transfer mode <MODE>
```

```
no flash call-transfer mode
```

Parameters

<MODE> – call transfer mode, may take the following values:

- r4 – call transfer is enabled after pressing R 4;
- hook – call transfer is enabled after going on-hook;
- both – call transfer is enabled after going on-hook and pressing R 4.

Default value

r4

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

CONFIG-VOIP-PROFILE

Example:

```
esr(config-voice-port-fxs)# flash call-transfer mode hook
```

flash timer

The command sets the minimum time of flash signal (flash call transfer) detection.

The use of a negative form (no) of the command returns the default value.

Syntax

```
flash timer <TIME>
no flash timer
```

Parameters

<TIME> – minimum time of flash signal detection in milliseconds, takes the values of [80..1000]

Default value

100

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

CONFIG-VOIP-PROFILE

Example:

```
esr(config-voice-port-fxs)# flash timer 200
```

hotline

The command enables *'Hot/Warm line'* service. This service allows to establish an outgoing connection automatically without dialling the number right after the lifting of a headset – «hot line», or with a delay – «warm line».

Syntax

```
[no] hotline
```

Parameters

The command does not contain parameters.

Default value

'Hot/Warm line' service is disabled.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# hotline
```

hotline ipt

The command enables 'Hot/Warm line' service in the direction from analogue telephone line to VoIP.

The use of a negative form (no) of the command disables 'Hot/Warm line' service in the direction from analogue telephone line to VoIP.

Syntax

```
[no] hotline ipt
```

Parameters

The command does not contain parameters.

Default value

'Hot/Warm line' service is disabled.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# hotline ipt
```

hotline pstn

The command enables 'Hot/Warm line' service in the direction from VoIP to PSTN.

The use of a negative form (no) of the command disables 'Hot/Warm line' service in the direction from VoIP to PSTN.

Syntax

```
[no] hotline pstn
```

Parameters

The command does not contain parameters.

Default value

'Hot/Warm line' service is disabled.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# hotline pstn
```

hotline number

The command assigns phone number for the connection establishment when using 'Hot/Warm line' service.

The use of a negative form (no) of the command removes 'Hot/Warm line' service number.

Syntax

```
hotline number <PHONE>
```

```
no hotline number
```

Parameters

<PHONE> – phone number for the connection establishment after a phone is picked up, set by the string from 1 to 50 characters.

Default value

The value is not specified.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# hotline number 5462
```

hotline number ipt

The command assigns phone number for the connection establishment when using *'Hot/Warm line'* service in the direction from analogue telephone line to VoIP.

The use of a negative form (no) of the command removes the number of *'Hot/Warm line'* service in the direction from analogue telephone line to VoIP.

Syntax

```
hotline number ipt <PHONE>
```

```
no hotline number ipt
```

Parameters

<PHONE> – phone number which is called when using «Hot/Warm line» service in the direction from analogue telephone line to VoIP, set by the string from 1 to 50 characters.

Default value

The value is not specified.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# hotline number ipt 6347
```

hotline number pstn

The command assigns phone number for the connection establishment when using *'Hot/Warm line'* service in the direction from VoIP to analogue telephone line.

The use of a negative form (no) of the command removes *'Hot/Warm line'* service number in the direction from VoIP to analogue telephone line.

Syntax

```
hotline number pstn <PHONE>
no hotline number pstn
```

Parameters

<PHONE> – hot number for the call in PSTN direction, set by the string of up to 50 characters.

Default value

The value is not specified.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# hotline number pstn 5462
```

hotline timeout

The command sets *'Hotline timeout'* before establishing the connection and after picking up a phone.

The use of a negative form (no) of the command returns the default value.

Syntax

```
hotline timeout <TIME>
no hotline timeout
```

Parameters

<TIME> – time interval in seconds within which connection with an opposing subscriber is established, takes the values of [0..60].

Default value

0

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# hotline timeout 10
```

hotline timeout ipt

The command sets 'Hotline timeout' before establishing the connection with 'Hot number' in the direction from analogue telephone line to VoIP.

The use of a negative form (no) of the command sets the default value.

Syntax

```
hotline timeout ipt <TIME>
no hotline timeout ipt
```

Parameters

<TIME> – time interval in seconds within which connection with an opposing subscriber is established, takes the values of [0..60].

Default value

0

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# hotline timeout ipt 10
```

hotline timeout pstn

The command sets 'Hotline timeout' before establishing the connection with 'Hot number' in the direction from VoIP to analogue telephone line.

The use of a negative form (no) of the command sets the default value.

Syntax

```
hotline timeout pstn <TIME>
no hotline timeout pstn
```

Parameters

<TIME> – time interval in seconds within which connection with an opposing subscriber is established, takes the values of [0..60].

Default value

0

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# hotline timeout pstn 10
```

hybrid rx

The command ensures incoming signal amplification.

Syntax

`hybrid rx <NUM>`

`no hybrid rx`

Parameters

<NUM> – takes values of [-200..200]

Default value

0

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# hybrid rx 100
```

hybrid tx

The command ensures outgoing signal amplification.

Syntax

`hybrid rx <NUM>`

`no hybrid rx`

Parameters

<NUM> – takes values of [-200..200]

Default value

0

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# hybrid rx 20
```

ipt prefix-name

The command specifies the prefix of CallerID name which is transmitted in VoIP direction.

The use of a negative form (no) of the command removes a specified value.

Syntax

`ipt prefix-name <NAME>`

`no ipt prefix-name`

Parameters

<NAME> – prefix that is added to CallerID name, set by the string from 1 to 21 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# ipt prefix-name localPSTN
```

ipt prefix-number

The command specifies the prefix of CallerID number which is transmitted in VoIP direction.

The use of a negative form (no) of the command removes a specified value.

Syntax

```
ipt prefix-number <NUMBER>
no ipt prefix-number
```

Parameters

<NUMBER> – prefix that is added to number in CallerID, set by the string from 1 to 21 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# ipt prefix-number 7
```

ipt offhook-ringing

The command sets the loop closure when calling from TDM to IP before voice channel forwarding in the VoIP direction.

The use of a negative form (no) of the command allows you not to close the loop when calling from TDM to IP until the vocal tract to SIP gateway, that interacts via the protocol, is received. The setting is used only in combination with 'Hotline' setting.

Syntax

```
[no] ipt offhook-ringing
```

Parameters

The command does not contain parameters.

Default value

Allows you not to close the loop when calling from TDM to IP before voice channel forwarding.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# no ipt offhook-ringing
```

ipt ring-number

The command sets the number of 'Call sendings' according to which FXO set will close the loop ('picks up the phone') and return a dial tone to phone line.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipt ring-number <COUNT>
no ipt ring-number
```

Parameters

<COUNT> – amount of 'Call sendings' according to which FXO set will close the loop, takes the value of [2..10].

Default value

2

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# ipt ring-number 4
```

profile sip

The command allows choosing a SIP profile for a port being configured.

Syntax

```
profile sip <PROFILE>
```

Parameters

<PROFILE> – SIP profile index, can take values [1..5].

Default value

None

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxo)# profile sip 1
```

profile pbx

The command allows choosing a SIP profile for a port being configured.

Syntax

`profile pbx<PROFILE>`

Parameters

<PROFILE> – name of the SIP profile created on the PBX server, set by a string from 1 to 31 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxo)# profile pbx fxs_ports
```

pstn transmit-number

The command sets the transmission of a number received from IP (from Request URI header of INVITE request) to the line, excluding port number.

The use of a negative form (no) of the command disables the transmission of number received from IP.

Syntax

```
[no] pstn transmit-number
```

Parameters

The command does not contain parameters.

Default value

Not to transmit full number.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# pstn transmit-number
```

pstn transmit-prefix

The command sets the transmission of FXO set subscriber number.

The use of a negative form (no) of the command disables the transmission of FXO set subscriber number.

Syntax

```
[no] pstn transmit-prefix
```

Parameters

The command does not contain parameters.

Default value

Transmit FXO set’s subscriber number.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# pstn transmit-prefix
```

shutdown

The command disables a port.

The use of a negative form (no) of the command enables the port.

Syntax

`[no] shutdown`

Parameters

The command does not contain parameters.

Default value

Port enabled.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# shutdown
```

sip port

The command specifies UDP port number to receive SIP messages coming to the account as well as to send outgoing SIP messages from the account.

The use of a negative form (no) of the command returns the default value.

Syntax

```
sip port <PORT>
```

```
no sip port
```

Parameters

<PORT> – UDP port number, takes values of [1..65535].

Default value

5060

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# sip port 5080
```

sip user display-name

The command allows you to specify user name matched with the port (displayed in ‘Display-Name’ field of ‘From’ header in outgoing SIP messages).

The use of a negative form (no) of the command removes the value.

Syntax

```
sip user display-name <LOGIN>
```

```
no sip user display-name
```

Parameters

<LOGIN> – user name displayed in the Display-Name field, set by the string of up to 31 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# sip user display-name "port-1"
```

sip user phone

The command assigns a subscriber number reserved for a telephone port.

The use of a negative form (no) of the command removes a subscriber number.

Syntax

```
sip user phone <PHONE>
no sip user phone
```

Parameters

<PHONE> – subscriber number reserved for a telephone port, set by the string of up to 50 characters.

Default value

The value is not specified.

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

CONFIG-VOICE-PORT-FXS

Example:

```
esr(config-voice-port-fxs)# sip user phone 4101
```

timing delay

The command sets the delay before dialling.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timing delay <TIME>
no timing delay
```

Parameters

<TIME> – delay before dialling in seconds, takes the values of [0..10].

Default value

2

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# timing delay 4
```

timing digit

The command sets a minimally detectable interdigit interval for FXS port.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timing digit <TIME>
no timing pulse-interdigit
```

Parameters

<TIME> – minimum interdigit interval in milliseconds, takes values of [150..20000].

Default value

200

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXS

CONFIG-VOIP-PROFILE

Example:

```
esr(config-voice-port-fxs)# timing interdigit 300
```

timing flash

The command sets time of loop closure to simulate 'flash' pulse.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timing flash <TIME>
no timing flash
```

Parameters

<TIME> – time of loop closure in milliseconds, takes values of [70..1000].

Default value

100

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# timing flash 150
```

timing pulse-digit

The command sets the impulse duration under pulse dialing.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timing pulse-digit <TIME>
no timing pulse-digit
```

Parameters

<TIME> – signal pulse width during pulse dialing in milliseconds, takes the values of [50..120].

Default value

80

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# timing pulse-digit 75
```

timing pulse-interdigit

The command sets interdigit interval when pulse dialing to analogue telephone line for FXO port.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timing pulse-interdigit <TIME>
no timing pulse-interdigit
```

Parameters

<TIME> – interdigit interval during pulse dialing in milliseconds, takes the values of [80..2500].

Default value

200

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# timing pulse-interdigit 300
```

timing pulse-pause

The command sets the pause duration under pulse dialing.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timing pulse-pause <TIME>
no timing pulse-pause
```

Parameters

<TIME> – pause duration when pulse dialing in milliseconds, takes the values of [50..100].

Default value

80

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# timing pulse-pause 75
```

timing tone-digit

The command sets the tone duration under pulse dialing.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timing tone-digit <TIME>
no timing tone-digit
```

Parameters

<TIME> – tone duration with DTMF dialing in milliseconds, takes the values of [65..100].

Default value

80

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# timing tone-digit 75
```

timing tone-interdigit

The command sets the pause duration under DTMF dialing.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timing tone-interdigit <TIME>
no timing tone-interdigit
```

Parameters

<TIME> – pause duration with DTMF dialing in milliseconds, takes the values of [80..2500].

Default value

100

Required privilege level

10

Command mode

CONFIG-VOICE-PORT-FXO

Example:

```
esr(config-voice-port-fxo)# timing tone-interdigit 150
```

Phone port testing

> ⚠ This functionality is supported only by ESR-12V/12VF/14VF routers

test voice-port start

The command starts the procedure of subscriber line electric performance measurement.

Syntax

`test voice-port <NUM> start`

Parameters

<NUM> – FXS proxy number, takes values of [1..3].

Required privilege level

1

Command mode

ROOT

Example:

```
esr# test voice-port 3 start
Voice-port 3 test started. The result will be available in 80 seconds, or more if the test run
on other ports.
```

test voice-port status

The command displays the results of last procedure of subscriber line electrical performance measurement.

Syntax

`test voice-port <NUM> status`

Parameters

<NUM> – FXS proxy number, takes values of [1..3].

Required privilege level

1

Command mode

ROOT

Example:

```
esr# test voice-port 3 status
Testing voice-port 3 is idle
Last test start: Sat May 20 16:01:37 2017
Number voip-port 3
  Foreign DC voltage A (TIP):    0.128377 U
  Foreign DC voltage B (RING):   0.144342 U
  Foreign AC voltage A (TIP):    0.026239 U
  Foreign AC voltage B (RING):   0.032287 U
  Cross current:                 0.260343 mA
  Longitudinal current:          -0.101857 mA
  Line supply voltage:           -50.370598 U
  Resistance A (TIP) - B (RING): 1007.203674 kOm
  Resistance A (TIP) - Ground:   402.105469 kOm
  Resistance B (RING) - Ground:  874.041443 kOm
  Capacity A (TIP) - B (RING):   50.000000 nF
  Capacity A (TIP) - Ground:     573.000000 nF
```

13 Tunnels configuration and monitoring

- auth-nocache
- authentication algorithm
- authentication method
- certificate
- clear ip nhrp
- clear tunnels counters
- clear tunnels softgre
- compression
- default-profile
- description
- dscp
- enable
- encryption algorithm
- history statistics
- ignore-default-route
- interface
- ip nhrp authentication
- ip nhrp enable
- ip nhrp holding-time
- ip nhrp ipsec
- ip nhrp map
- ip nhrp multicast
- ip nhrp nhs
- ip nhrp redirect
- ip nhrp shortcut
- ip tcp adjust-mss
- ipsec authentication method
- ipsec authentication pre-shared-key
- keepalive dhcp dependent-interface
- keepalive dhcp link-timeout
- keepalive dst-address
- keepalive enable
- keepalive retries
- keepalive timeout
- keepalive timeout ipsec
- key
- load-average
- local address
- local address xauth
- local checksum
- local cookie
- local interface
- local port
- local session-id
- mode
- mode
- mtu
- multipoint
- password
- peer lt
- ppp failure-count
- ppp timeout keepalive
- protocol

- protocol
- remote address
- remote address
- remote address xauth
- remote checksum
- remote cookie
- remote port
- remote session-id
- route-metric
- route-nopull
- snmp init-trap
- show ip nhrp
- show tunnels configuration
- show tunnels counters
- show tunnels history
- show tunnels status
- show tunnels utilization
- ttl
- tunnel
- tunnel
- tunnel-source
- username
- username

auth-nocache

This command disables the password caching of the OPENVPN client user.

The use of a negative form (no) of the command activates caching the password of the OPENVPN client user.

Syntax

```
[no] auth-nocache
```

Parameters

None.

Default value

Caching is allowed.

Required privilege level

15

Command mode

CONFIG-OPENVPN

Example

```
esr(config)# auth-nocache
```

authentication algorithm

This command sets the authentication algorithm, which is used for authentication when connecting to the OPENVPN server.

The use of a negative form (no) of the command removes the authentication algorithm.

Syntax

```
authentication algorithm <ALGORITHM>
no authentication algorithm
```

Parameters

<ALGORITHM> – authentication algorithm, may take values: md4, rsa-md4, md5, rsa-md5, mdc2, rsa-mdc2, sha, sha1, rsa-sha, rsa-sha1, rsa-sha1-2, dsa, dsa-sha, dsa-sha1, dsa-sha1-old, ripemd160, rsa-ripemd160, ecdsa-with-sha1, sha-224, rsa-sha-224, sha-256, rsa-sha-256, sha-384, rsa-sha-384, sha-512, rsa-sha-512, whirlpool.

Default value

Unspecified.

Required privilege level

15

Command mode

CONFIG-OPENVPN

Example

```
esr(config-openvpn)# authentication algorithm md5
```

authentication method

The command specifies authentication method which will be used when establishing a remote connection by PPPoE, PPTP and L2TP clients.

The use of a negative form (no) of the command removes a specified authentication method.

Syntax

```
authentication method <METHOD>
```

```
no authentication method <METHOD>
```

Parameters

<METHOD> – authentication method, possible values: chap, mschap, mschap-v2, eap, pap.

Default value

chap

Required privilege level

10

Command mode

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-L2TP

Example

```
esr(config-pppoe)# authentication method mschap-v2
```

certificate

This command specifies the necessary certificates for connecting to the OPENVPN server.

The use of a negative form (no) of the command removes certificate name from the configuration.

Syntax

```
certificate <CERTIFICATE-TYPE> <NAME>
no certificate <CERTIFICATE-TYPE>
```

Parameters

<CERTIFICATE-TYPE> – certificate or key type, may take the following values:

- ca – certificate authority certificate;
- client-crt – client certificate;
- client-key – client key;
- crl – Certificate Revocation List;
- dh – Diffie-Hellman key;
- ta – HMAC key.

<NAME> – certificate or key name, set by the string of up to 31 characters.

Default value

None

Required privilege level

15

Command mode

CONFIG-OPENVPN

Example

```
esr(config-openvpn)# certificate ca KEY
```

clear ip nhrp

This command clears NHRP records from the router.

Syntax

```
clear ip nhrp [ { dynamic | static | incomplete | nhs} ] [ { nbma-address <ADDR> | tunnel gre <ID> | tunnel-address <ADDR> } ] [ vrf <VRF> ]
```

Parameters

<ID> – tunnel identifier;

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<VRF> – VRF instance name, set by the string of up to 31 characters. When this parameter is specified, the NHRP records specified in this VRF will be deleted;

dynamic – clear dynamically acquired information;

incomplete – clear blank entries;

nbma-address – clear entries regarding a specific NBMA address;

nhs – clear entries regarding a specific NHS;

static – clear statically set records;

tunnel – clear entries regarding a specific tunnel;

tunnel-address – clear entries regarding a specific tunnel;

vrf – clear records in a specific VRF.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear ip nhrp vrf vrf_test tunnel gre 9 static
```

clear tunnels counters

The command performs the reset of specified tunnel/tunnel group counters.

Syntax

```
clear tunnels counters [ <TUN> ]
```

Parameters

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

You can specify several tunnels separated by commas ',' or you can specify the range of ports with '-'. If tunnel indexes are not specified, then the counters of all tunnels of a specified group will be cleared.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear tunnels counters gre 25
```

clear tunnels softgre

This command breaks the softgre tunnel/tunnels.

Syntax

```
clear tunnels softgre [ remote-address <REMOTE-IP> ]
```

Parameters

<REMOTE-IP> – remote IP address from which the softgre-tunnel was set.

Required privilege level

15

Command mode

ROOT

Example

```
esr# clear tunnels softgre esr#clear tunnels softgre remote-address10.10.42.10
```

compression

This command enables the mechanism of transmitted data compression between clients and the OPENVPN server.

The use of a negative form (no) of the command disables the mechanism of transmitted data compression.

Syntax

```
[no] compression
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-OPENVPN

Example

```
esr(config-openvpn)# compression
```

default-profile

The command allows you to use the SoftGRE tunnel configuration to automatically create tunnels with the same mode and local address.

The use of a negative form (no) of the command prohibits the use of tunnel configuration to automatically create tunnels.

Syntax

```
[no] default-profile
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-SOFTGRE

Example

```
esr(config-softgre)# default-profile
```

description

The command is used to change a description of configured tunnel.

The use of a negative form (no) of the command removes a specified description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – tunnel description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-IP4IP4

CONFIG-SUBTUNNEL

CONFIG-GRE

CONFIG-L2TP

CONFIG-L2TPV3

CONFIG-VTI

CONFIG-LT

CONFIG-PPTP

CONFIG-PPPOE

CONFIG-OPENVPN

Example

```
esr(config-gre)# description "tunnel to branch"
```

dscp

The command sets the DSCP code value for the use in IP headers of encapsulate packets.

The use of a negative form (no) of the command sets the default DSCP value.

Syntax

```
dscp <DSCP>
no dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Default value

Inherited from encapsulated packet.

Required privilege level

10

Command mode

CONFIG-IP4IP4

CONFIG-GRE

Example

```
esr(config-ip4ip4)# dscp 40
```

enable

The command enables a tunnel.

The use of a negative form (no) of the command disables the tunnel.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

Tunnel is disabled.

Required privilege level

10

Command mode

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TP

CONFIG-L2TPV3

CONFIG-VTI

CONFIG-LT

CONFIG-PPTP

CONFIG-PPPOE

CONFIG-OPENVPN

Example

```
esr(config-gre)# enable
```

encryption algorithm

This command selects the encryption algorithm used when data transmission.

The use of a negative form (no) of the command disables the encryption.

Syntax

```
encryption algorithm <ALGORITHM>
no encryption algorithm
```

Parameters

<ALGORITHM> – encryption protocol identifier, may take following values: des, blowfish128, aes128, des-ede, aes192, 3des, desx, aes256.

Default value

Encryption disabled.

Required privilege level

15

Command mode

CONFIG-OPENVPN

Example

```
esr(config-openvpn)# encryption algorithm aes128
```

history statistics

The command enables maintenance of a current tunnel usage statistics.

The use of a negative (no) form of the command disable maintenance of a current tunnel usage statistics.

Syntax

`[no] history statistics`

Parameters

None.

Required privilege level

10

Command mode

CONFIG-GRE

CONFIG-SOFTGRE

CONFIG-SUBTUNNEL

CONFIG-IP4IP4

CONFIG-L2TPV3

CONFIG-LT

CONFIG-VTI

CONFIG-PPTP

CONFIG-PPPOE

CONFIG-OPENVPN

Example

```
esr(config-ip4ip4)# history statistics
```

ignore-default-route

The command enables the mode in which a default route received from the server is not set in the routing table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] ignore-default-route
```

Parameters

The command does not contain parameters.

Default value

A default route received from the server is set in the routing table.

Required privilege level

10

Command mode

CONFIG-L2TP

CONFIG-PPTP

CONFIG-PPPOE

Example

```
esr(config-pptp)# ignore-default-route
```

interface

The command specifies the interface for PPPoE connection establishing.

The use of a negative form (no) of the command removes a specified interface.

Syntax

```
interface <IF>
no interface
```

Parameters

<IF> – an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces.

Required privilege level

10

Command mode

CONFIG-PPPOE

Example

```
esr(config-pppoe)# interface gigabitethernet 1/0/5.100
```

ip nhrp authentication

The command enables authentication for NHRP protocol. All participants in the NHRP process must have the same password.

The use of a negative form (no) of the command disables the authentication.

Syntax

```
ip nhrp authentication <WORD>
[no] ip nhrp authentication
```

Parameters

<WORD> – unencrypted password, set by the string of [1..8] characters, may include [0-9a-fA-F] characters.

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# ip nhrp authentication pass
```

ip nhrp enable

This command enables NHRP in the router tunnel.

The use of a negative form (no) of the command disables the NHRP protocol on the router.

Syntax

```
[no] ip nhrp enable
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# ip nhrp enable
```

ip nhrp holding-time

This command sets the time during which a client record will exist on the NHRP server. This command also sets the frequency of client sending requests for registration to the NHRP server; it is equal to 1/3 of NHRP holding time.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip nhrp holding-time <TIME>
[no] ip nhrp holding-time
```

Parameters

<TIME> – the time in seconds during which a record about this client will exist on the server takes the values [1..65535].

Default value

7200 seconds

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# ip nhrp holding-time
```

ip nhrp ipsec

This command specifies the use of the previously created IPsec-VPN on the mGRE tunnel.

The use of a negative form (no) of the command disables the use of IPsec-VPN with the mGRE tunnel.

Syntax

```
ip nhrp ipsec <WORD> { static | dynamic }
no ip nhrp ipsec <WORD> { static | dynamic }
```

Parameters

<WORD> – VPN name, set by the string of up to 31 characters.

static – static connection, used to communicate with the NHRP server, set on the client, unlimited in time;

dynamic – dynamically established connection, configured to communicate with the NHRP client.

Required privilege level

15

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# ip nhrp ipsec VPN static
```

ip nhrp map

This command sets the match between 'internal' tunnel address and the 'external' NBMA address.

The use of a negative form (no) of the command removes the match.

Syntax

```
[no] ip nhrp map <ADDR-IN> <ADDR-OUT>
```

Parameters

<ADDR-IN> –tunnel interface IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<ADDR-OUT> – external interface IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# ip nhrp map 192.168.1.2 67.86.141.231
```

ip nhrp multicast

This command defines the destination of multicast traffic.

The use of a negative form (no) of the command removes the destination.

Syntax

```
[no] ip nhrp multicast { dynamic | nhs | <ADDR> }
```

Parameters

dynamic — sends traffic to all peers with which there is a connection;

nhs — sends to all static configured NHRP servers;

<ADDR> – sends to specifically configured IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# ip nhrp multicast nhs
```

ip nhrp nhs

This command is used to set the 'logical (tunnel)' address of the NHRP server.

The use of a negative form (no) of the command removes the entry about server.

Syntax

```
ip nhrp nhs <ADDR/LEN> [ no-registration ]
no ip nhrp nhs <ADDR/LEN>
```

Parameters

<ADDR/LEN> – address, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32];

no-registration — do not register on the NHRP server.

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# ip nhrp nhs 192.168.1.2
```

ip nhrp redirect

This command enables the mode in which the NHRP server can send NHRP Traffic Indication messages.

The use of a negative form (no) of the command disables the mode.

Syntax

```
[no] ip nhrp redirect
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# ip nhrp redirect
```

ip nhrp shortcut

This command enables a mode that allows the use of the shortest routes for communication with other clients.

The use of a negative form (no) of the command disables the mode.

Syntax

```
[no] ip nhrp shortcut
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# ip nhrp shortcut
```

ip tcp adjust-mss

This command overrides the value of the MSS (Maximum segment size) field in incoming TCP packets.

The use of a negative form (no) of the command disables MSS field value correction.

Syntax

```
ip tcp adjust-mss <MSS>
no ip tcp adjust-mss
```

Parameters

<MSS> – MSS value, takes values in the range of [500..1460].

Default value

1460

Required privilege level

10

Command mode

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-VTI

CONFIG-LT

Example

```
esr(config-gre)# ip tcp adjust-mss 1400
```

ipsec authentication method

This command selects the key authentication method for the IKE connection. Message authentication by key is used when an IKE connection is established. The key is set by the 'ipsec authentication pre-shared-key' command (see section ipsec authentication pre-shared-key).

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipsec authentication method pre-shared-key
no ipsec authentication method
```

Parameters

pre-shared-key – authentication method using pre-received encryption keys.

Required privilege level

10

Command mode

CONFIG-L2TP

Example

```
esr(config-l2tp-server)# ipsec authentication method psk
```

ipsec authentication pre-shared-key

This command specifies a shared secret authentication key that should be the same for both parties of the tunnel.

The use of a negative form (no) of the command removes a set key.

Syntax

```
ipsec authentication pre-shared-key { ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }|
hexadecimal {<HEX> | encrypted <ENCRYPTED-HEX> } }
```

```
no ipsec authentication pre-shared-key
```

Parameters

<TEXT> – string [1..64] ASCII characters.

<HEX> – number, [1..32] bytes size, set by the string of [2..128] characters in hexadecimal format (0xYYYY ...) or (YYYY ...).

<ENCRYPTED-TEXT> – encrypted password, [1..32] bytes size, set by the string of [2..128] characters.

<ENCRYPTED-TEXT> – encrypted number, [2..64] bytes size, set by the string of [2..256] characters.

Default value

none

Required privilege level

10

Command mode

CONFIG-L2TP

Example

```
esr(config-l2tp-server)# ipsec authentication pre-shared-key ascii-text password
```

keepalive dhcp dependent-interface

The command enables the mechanism of IP addresses iterative query using DHCP on the specified interfaces when the GRE tunnel is disconnected via keepalive. You can specify up to 8 interfaces for each GRE tunnel.

The use of a negative form (no) of the command disables the mechanism of IP addresses iterative query using DHCP.

Syntax

```
keepalive dhcp dependent-interface <IF>
```

```
no keepalive dst-address
```

Parameters

<IF> – physical or aggregated interface, specified in the form described in Section Types and naming order of router interfaces.

Default value

None

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# keepalive dhcp dependent-interface gi 1/0/1
```

keepalive dhcp link-timeout

The command specifies the time interval between GRE tunnel disabling and IP address iterative query on the interface/interfaces specified by the keepalive dhcp dependent-interface command (see Section keepalive dhcp dependent-interface)

The use of a negative form (no) of the command sets the default value.

Syntax

```
keepalive dhcp link-timeout <SEC>
no keepalive dhcp link-timeout
```

Parameters

<SEC> – interval in seconds between GRE tunnel disabling and IP address iterative query on the interface/interfaces specified by the keepalive dhcp dependent-interface command, takes the value of [1..32767] seconds.

Default value

10

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# keepalive dhcp link-timeout 90
```

keepalive dst-address

The command configures IP address to send keepalive packets for checking tunnel capability. If a specified IP address is not available, the tunnel switches the operational status to DOWN. The parameter has a value only with keepalive mechanism enabled (see Section keepalive enable).

The use of a negative form (no) of the command disables the checking.

Syntax

```
keepalive dst-address <ADDR>
no keepalive dst-address
```

Parameters

<ADDR> – IP address to check GRE tunnel capability.

Default value

None

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# keepalive dst-address 192.168.1.57
```

keepalive enable

The command enables the checking of remote tunnel gateway availability. If a remote tunnel gateway is not available, the tunnel switches the operational status to DOWN.

The use of a negative form (no) of the command disables the checking.

Syntax

```
[no] keepalive enable
```

Parameters

The command does not contain parameters.

Default value.

Disabled.

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# keepalive enable
```

keepalive retries

The command defines the number of attempts to check the remote tunnel gateway availability. Upon reaching the specified number of failed attempts, the tunnel will be considered inactive.

The use of a negative form (no) of the command sets the default value.

Syntax

```
keepalive retries <VALUE>
no keepalive retries
```

Parameters

<VALUE> – number of attempts, takes values in the range of [1..255].

Default value

6

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# keepalive retries 8
```

keepalive timeout

The command controls the period of sending keepalive packets to opposing party.

The use of a negative form (no) of the command sets the default value.

Syntax

```
keepalive timeout <TIME>
no keepalive timeout
```

Parameters

<TIME> – time in seconds, takes values of [1..32767].

Default value

10

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# keepalive timeout 18
```

keepalive timeout ipsec

This command sets the time that is allowed to restore the IPsec VPN connection, after which the router will reboot. This functionality only works when the router is in OTT mode. The operating mode of the OTT is enabled when the device is manufactured.

The use of a negative form (no) of the command reduces the value to the default – 180.

Syntax

```
keepalive timeout ipsec <TIME>
no keepalive timeout ipsec
```

Parameters

<TIME> – time in seconds, takes values of [30..32767].

Default value

180

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# keepalive timeout ipsec 1000
```

key

The command enables key transmission in GRE tunnel header (according to RFC 2890) and sets the key value. The key can be used to identify traffic streams in GRE tunnel.

The use of a negative form (no) of the command disables key transmission.

Syntax

```
key <KEY>
no key
```

Parameters

<KEY> – KEY value, takes values in the range of [1..2000000].

Default value

Key is not transmitted.

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# key 40
```

load-average

The command specifies the time interval during which the statistics on tunnel load is averaged.

The use of a negative form (no) of the command sets the default value.

Syntax

`load-average <TIME>`

`no load-average`

Parameters

<TIME> – interval in seconds, takes values of [5..150].

Default value

5

Required privilege level

10

Command mode

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

CONFIG-SUBTUNNEL

CONFIG-L2TPv3

CONFIG-VTI

CONFIG-L2TP

CONFIG-PPTP

CONFIG-PPPOE

CONFIG-OPENVPN

Example

```
esr(config-gre)# load-average
```

local address

The command sets IP address of a local tunnel gateway.

The use of a negative form (no) of the command removes local gateway IP address.

Syntax

```
local address <ADDR>
no local address
```

Parameters

<ADDR> – IP address of a local gateway.

Required privilege level

10

Command mode

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TPV3

CONFIG-VTI

Example

```
esr(config-ip4ip4)# local address 192.168.1.1
```

local address xauth

This command sets the use of the address issued by mode config when using the previously configured IPsec VPN in the XAUTH client mode.

The use of a negative form (no) of the command removes the configuration.

Syntax

```
local address xauth <NAME>
no local address
```

Parameters

<NAME> – name of the created before IPsec VPN, set by the string of up to 31 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-)# local address xauth IPsecVPN
```

local checksum

The command enables the calculation of the checksum and entry it to the GRE header of the packets to be sent.

The use of a negative form (no) of the command disables the calculation and sending of the checksum.

Syntax

`[no] local checksum`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# local checksum
```

local cookie

The command defines cookie value to check the conformance of data being transmitted and session.

The use of a negative form (no) of the command removes a local cookie.

Syntax

`local cookie <COOKIE>`

`no local cookie`

Parameters

<COOKIE> – COOKIE value, the parameter takes values of 8 or 16 characters in hexadecimal form.

Required privilege level

10

Command mode

CONFIG-L2TPV3

Example

```
esr(config-l2tpv3)# local cookie 8FB51B8FB
```

local interface

The command sets the use of IP address assigned to the interface as a GRE tunnel local gateway.

The use of a negative form (no) of the command stops the use of IP address assigned to the interface as a local gateway.

Syntax

```
local interface { <IF> | <TUN> }
no local interface
```

Parameters

<IF> – interface type and identifier specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# local interface gigabitethernet 1/0/1
```

local port

The command defines local UDP port if UDP was selected as encapsulation method.

The use of a negative form (no) of the command removes a local UDP port number.

Syntax

```
local port <UDP>
no local port
```

Parameters

<UDP> – UDP port number in the range of [1..65535].

Required privilege level

10

Command mode

CONFIG-L2TPV3

Example

```
esr(config-l2tpv3)# local port 1501
```

local session-id

Set local session identifier.

The use of a negative form (no) of the command removes a local session identifier.

Syntax

```
local session-id <SESSION-ID>
no local session-id
```

Parameters

<SESSION-ID> – session identifier, takes values in the range of [1..200000].

Required privilege level

10

Command mode

CONFIG-L2TPV3

Example

```
esr(config-l2tpv3)# local session-id 200
```

mode

The command sets SoftGRE tunnel operation mode.

The use of a negative form (no) of the command disables a set mode.

Syntax

```
mode <MODE>
no mode
```

Parameters

<MODE> – tunnel operation mode, takes the following values:

- data – data mode;
- management – management mode.

Required privilege level

10

Command mode

CONFIG-SOFTGRE

Example

```
esr(config-softgre)# mode data
```

mode

The command specifies the encapsulation mode for GRE tunnel.

The use of a negative form (no) of the command sets the default encapsulation.

Syntax

```
mode <MODE>
```

Parameters

<MODE> – GRE tunnel encapsulation mode:

- ip – encapsulation of IP packets in GRE;
- ethernet – encapsulation of Ethernet frames in GRE.

Default value

ip

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# mode ethernet
```

mtu

The command specifies MTU (Maximum Transmission Unit) size for tunnels

The use of a negative form (no) of the command sets the default MTU value.

Syntax

```
mtu <MTU>
no mtu
```

Parameters

<MTU> – MTU value, takes values in the range of [552..10000].

Default value

1500

Required privilege level

10

Command mode

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TP

CONFIG-L2TPV3

CONFIG-VTI

CONFIG-LT

CONFIG-PPTP

CONFIG-PPPOE

CONFIG-OPENVPN

Example

```
esr(config-l2tpv3)# mtu 1400
```

multipoint

This command puts the tunnel in multipoint mode. In this mode, it is possible to establish several connections from one tunnel interface.

The use of a negative form (no) of the command puts it into normal point-to-point mode.

Syntax

```
[no] multipoint
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# multipoint
```

password

This command sets the user password for the OPENVPN server.

The use of a negative form (no) of the command removes a user’s password.

Syntax

```
password { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no password
```

Parameters

<CLEAR-TEXT> – unencrypted password, set by the string of [8..32] characters, may include [0-9a-fA-F] characters;

<ENCRYPTED-TEXT> – unencrypted password, set by the string of [8..32] characters.

> The password are kept in encrypted form independently of format used when entering the command.

Required privilege level

15

Command mode

CONFIG-OPENVPN

Example

```
esr(config-openvpn)# password 01234567
```

peer lt

The command specifies a remote party (in another VRF) of a logical tunnel.

The use of a negative form (no) of the command removes a bind of tunnel remote party.

Syntax

```
[no] peer lt <ID>
```

Parameters

<ID> – logical tunnel remote party identifier.

Required privilege level

10

Command mode

CONFIG-LT

Example

```
esr(config-lt)# peer lt 2
```

ppp failure-count

This command sets the number of failed data-link tests before breaking the session.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ppp failure-count <NUM>
no ppp failure-count
```

Parameters

<NUM> – the number of failed data-link tests, specified in the range [1..100].

Default value

10

Required privilege level

10

Command mode

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

Example

```
esr(config-l2tp)# ppp failure-count 20
```

ppp timeout keepalive

The command specifies the time interval in seconds after which the router sends a keepalive message.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ppp timeout keepalive <TIME >
no ppp timeout keepalive
```

Parameters

<TIME> – time in seconds, takes values of [1..32767].

Default value

10

Required privilege level

10

Command mode

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

Example

```
esr(config-l2tp)# ppp timeout keepalive 5000
```

protocol

Select encapsulation method for L2TPv3 tunnel.

Syntax

```
protocol <TYPE>
no protocol
```

Parameters

<TYPE> – encapsulation type, possible values:

- IP – encapsulation in IP packet;
- UDP – encapsulation in UDP datagrams.

Required privilege level

15

Command mode

CONFIG-L2TPV3

Example

```
esr(config-l2tpv3)# protocol ip
```

protocol

Select encapsulation method for OPENVPN tunnel.

Syntax

```
protocol <TYPE>
```

```
no protocol
```

Parameters

<TYPE> – encapsulation type, possible values:

- TCP – encapsulation in TCP segments;
- UDP – encapsulation in UDP datagrams.

Default value

TCP

Required privilege level

15

Command mode

CONFIG-OPENVPN

Example

```
esr(config-openvpn)# protocol tcp
```

remote address

The command sets IP address of a remote tunnel gateway.

The use of a negative form (no) of the command removes remote gateway IP address.

Syntax

```
remote address <ADDR>
```

```
no remote address
```

Parameters

<ADDR> – IP address of a remote gateway.

Required privilege level

10

Command mode

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-SOFTGRE

CONFIG-L2TP

CONFIG-L2TPV3

CONFIG-VTI

CONFIG-PPTP

Example

```
esr(config-ip4ip4)# remote address 192.168.1.2
```

remote address

This command sets the IP address and TCP/UDP port of the remote tunnel gateway.

The use of a negative form (no) of the command removes remote gateway IP address.

Syntax

```
[no] remote address <ADDR> [ port <PORT>]
```

Parameters

<ADDR> – remote gateway IP address;

<PORT> – number of remote gateway TCP/UDP port in the range of [1..65535].

Default value

<PORT> – 1194.

Required privilege level

10

Command mode

CONFIG-OPENVPN

Example

```
esr(config-openvpn)# remote address 192.168.1.2 port 1233
```

remote address xauth

This command sets the use of the management-ip or data-ip address issued by mode config when using the previously configured IPsec VPN in the XAUTH client mode. Requires appropriate settings on the IPsec-VPN server.

The use of a negative form (no) of the command removes the configuration.

Syntax

```
remote address xauth <NAME> {management-ip|data-ip}
```

```
no remote address
```

Parameters

<NAME> – name of the created before IPsec VPN, set by the string of up to 31 characters;

management-ip – address obtained by mode config when installing IPsec VPN in the XAUTH client mode. Requires an ELTEX_MANAGEMENT_IP (28683) in the router IPsec-VPN server settings.

data-ip – address obtained by mode config when installing IPsec VPN in the XAUTH client mode. Requires an ELTEX_DATA_IP (28684) in the router IPsec-VPN server settings.

Default value

Not specified.

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# remote address xauth IPsecVPN
```

remote checksum

The command enables verification of the presence and consistency of checksum values in the headers of GRE packets being received.

The use of a negative form (no) of the command disables the checksum verification..

Syntax

```
[no] remote checksum
```

Parameters

The command does not contain parameters.

Default value

The checksum verification is disabled by default.

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# remote checksum
```

remote cookie

The command defines cookie value to check the conformance of data being transmitted and session.

The use of a negative form (no) of the command removes a deleted cookie.

Syntax

```
remote cookie <COOKIE>
no remote cookie
```

Parameters

<COOKIE> – COOKIE value, the parameter takes values of 8 or 16 characters in hexadecimal form.

Required privilege level

10

Command mode

CONFIG-L2TPV3

Example

```
esr(config-l2tpv3)# remote cookie 8FB51B8FB
```

remote port

The command defines remote UDP port if UDP was selected as encapsulation method.

The use of a negative form (no) of the command removes a specified UDP port number.

Syntax

```
remote port <UDP>
no remote port
```

Parameters

<UDP> – UDP port number in the range of [1..65535].

Default value

None

Required privilege level

10

Command mode

CONFIG-L2TPV3

Example

```
esr(config-l2tpv3)# remote port 65000
```

remote session-id

The command sets remote session identifier.

The use of a negative form (no) of the command removes a remote session identifier.

Syntax

```
remote session-id <SESSION-ID>
no remote session-id
```

Parameters

<SESSION-ID> – session identifier, takes values in the range of [1..200000].

Required privilege level

10

Command mode

CONFIG-L2TPV3

Example

```
esr(config-l2tpv3)# remote session-id 2
```

route-metric

This command assigns the metric of routes received by the client from the OPENVPN server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
route-metric <METRIC>
no route-metric
```

Parameters

<METRIC> – route metric, takes values of [0..255].

Default value

0

Required privilege level

15

Command mode

CONFIG-OPENVPN

Example

```
esr(config)# route-metric 100
```

route-nopull

This command disables the use of routes transmitted by the OPENVPN server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] route-nopull
```

Parameters

None.

Default value

Route ignore is disabled.

Required privilege level

15

Command mode

CONFIG-OPENVPN

Example

```
esr(config)# route-nopull
```

snmp init-trap

The command enables sending snmp-trap about tunnel enabling/disabling.

The use of a negative form (no) of the command disables sending snmp-trap about tunnel enabling/disabling.

Syntax

`[no] snmp init-trap`

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-GRE

CONFIG-SUBTUNNEL

Example

```
esr(config-gre)# snmp init-trap
```

show ip nhrp

This command is used to view NHRP records.

Syntax

```
show ip nhrp [ { dynamic | static | incomplete | nhs } ] [ { nbma-address <ADDR> | tunnel gre <ID> | tunnel-address <ADDR> } ] [ vrf <VRF> ]
```

Parameters

<ID> – tunnel identifier;

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<VRF> – VRF instance name, specified by the string from 1 to 31 characters long. When this parameter is specified, the NHRP records specified in this VRF will be displayed;

dynamic – show information acquired dynamically;

incomplete – show blank entries;

nbma-address – show entries with a specific NBMA address;

nhs – show entries of a specific NHS;

static – show statically set records;

tunnel – show entries of a specific tunnel;

tunnel-address – show entries with a specific tunnel address;

vrf – show records in a specific VRF.

Required privilege level

10

Command mode

ROOT

Example

```
esr# hub-12# sh ip nhrp
Tunnel address     NBMA address       Interface   Peer type         Expire      Created
Flags
----------------   ----------------   ---------   ---------------   ---------   ---------
---------
12.1.1.3           192.168.3.4        gre 1       dynamic           0:04:38     00:23:18
lower-up,
                                                                                            up

12.1.1.9           192.168.9.4        gre 1       dynamic           0:04:45     00:23:49
lower-up,
                                                                                            up

12.1.1.99          120.12.120.12      gre 1       dynamic           0:01:14     00:24:11
lower-up,
                                                                                            up
```

show tunnels configuration

The command displays tunnel configuration.

Syntax

```
show tunnels configuration [ <TUN> ]
```

Parameters

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

Required privilege level

1

Command mode

ROOT

Example

```
esr# show tunnels configuration gre 25
State:                                             enabled
Description:
Local address:                                     14.0.0.2
Remote address:                                    14.0.0.1
Calculates checksums for outgoing GRE packets:     no
Requires that all input GRE packets were checksum: no
key:                                               -
TTL:                                               Inherit
DSCP:                                              0
MTU:                                               1500
Security zone:                                     remote
```

show tunnels counters

The command displays tunnel counters.

Syntax

```
show tunnels counters [ <TUN> ]
```

Parameters

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

You may specify several tunnels. If tunnel indexes are not specified, then the counters of all tunnels of a specified group will be displayed. If a certain tunnel is specified, the detailed information on this tunnel will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show tunnels counters l2tpv3 1
Tunnel 'l2tpv3 1' counters:
 Packets received:              0
 Bytes received:                0
 Dropped on receive:            0
 Receive errors:                0
 Multicasts received:           0
 Receive length errors:         0
 Receive buffer overflow errors: 0
 Receive CRC errors:            0
 Receive frame errors:          0
 Receive FIFO errors:           0
 Receive missed errors:         0
 Receive compressed:            0
 Packets transmitted:           658
 Bytes transmitted:             56588
 Dropped on transmit:           0
 Transmit errors:               0
 Transmit aborted errors:       0
 Transmit carrier errors:       0
 Transmit FIFO errors:          0
 Transmit heartbeat errors:     0
 Transmit window errors:        0
 Transmit comressed:            0
 Collisions:                    0
```

show tunnels history

The command is used to view tunnel usage statistics.

Syntax

```
show tunnels history [ <TUN> ] [<timer {TIMER}>]
```

Parameters

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

<TIMER > – timer optional key. Has the following parameters:

- hours displays history in the last 72 hours;
- minutes displays history in the last 60 hours;
- seconds displays history in the last 60 seconds;
- When timer is missing, 3 tables containing tunnel usage history are displayed.

Required privilege level

5

Command mode

ROOT

Example

```
esr# show tunnel history gre 1 timer minutes
gre 1
Last 60 minutes:
Timer  Recv utilization, Kbit/s  Sent utilization, Kbit/s  Recv errors  Sent errors  Output
drops
------ ------------------------- ------------------------- ------------ ------------
--------------
0-1   240            16             0       0       0
1-2   961            64             0       0       0
2-3   962            64             0       0       0
3-4   962            64             0       0       0
4-5   960            64             0       0       0
5-6   961            64             0       0       0
6-7   719            64             0       0       0
7-8   960            64             0       0       0
8-9   800            65             0       0       0
9-10  962            64             0       0       0
10-11 865            64             0       0       0
11-12 962            64             0       0       0
12-13 817            65             0       0       0
13-14 962            65             0       0       0
14-15 961            65             0       0       0
15-16 880            60             0       0       0
16-17 960            63             0       0       0
17-18  0             0              0       0       0
18-19  0             0              0       0       0
19-20  0             0              0       0       0
20-21  0             0              0       0       0
21-22  0             0              0       0       0
```

show tunnels status

The command is used to display system interfaces status.

Syntax

```
show tunnels status [ <TUN> ]
```

Parameters

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

You may specify several tunnels in command. If tunnel indexes are not specified, then the statuses of all tunnels of a specified group will be displayed. If a certain tunnel is specified, the detailed information on this tunnel will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show tunnels status
Tunnel     Admin state        MTU      Local IP           Remote IP          Uptime
------      ----------       -----    ------------       -----------        ---------------
ip4ip4 4         Up           1500     115.0.0.100        115.0.0.30         1 minute and 4
 seconds
```

show tunnels utilization

The command displays the average load in tunnels over the specified period.

Syntax

```
show tunnels utilization [ <TUN> ]
```

Parameters

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

You can specify several tunnels separated by commas ',' or you can specify the range of interfaces with '-'. If tunnel indexes are not specified, then the counters of all tunnels of a specified group will be cleared.

Required privilege level

10

Command mode

ROOT

Example

```
esr# show tunnels utilization gre 2
Tunnel       Period, s     Sent,         Recv,        Frames Sent   Frames Recv
                           Kbit/s        Kbit/s
---------    -----------   -----------   -----------   -----------   -----------
gre 2        15            0             0             0             0
```

ttl

The command specifies the TTL lifetime for tunnel packets.

The use of a negative form (no) of the command sets the default TTL value.

Syntax

`ttl <TTL>`

`no ttl`

Parameters

<TTL> – TTL value, takes values in the range of [1..255].

Default value

Inherited from encapsulated packet.

Required privilege level

10

Command mode

CONFIG-IP4IP4

CONFIG-GRE

Example

```
esr(config-ip4ip4)# ttl 10
```

tunnel

The command allows you to switch to the tunnel configuration mode.

The use of a negative form (no) of the command removes the tunnel.

Syntax

`[no] tunnel <TUN>`

Parameters

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

Required privilege level

10

Command mode

CONFIG

Example 1

Switch to tunnel l2tp 1/10/1 configuration mode:

```
esr(config)# tunnel l2tp 10
esr(config-l2tp)#
```

Example 2

Switch to tunnel l2tpv3 1/10/1 configuration mode:

```
esr(config)# tunnel l2tpv3 10
esr(config-l2tpv3)#
```

Example 3

Switch to tunnel ip4ip4 1/200/1 configuration mode:

```
esr(config)# tunnel ip4ip4 200
esr(config-ip4ip4)#
```

Example 4

Switch to tunnel gre 25 configuration mode:

```
esr(config)# tunnel gre 25
esr(config-gre)#
```

Example 5

Switch to tunnel vti 125 configuration mode:

```
esr(config)# tunnel vti 125
esr(config-vti)#
```

Example 6

Switch to tunnel pptp 10 configuration mode:

```
esr(config)# tunnel pptp 10
esr(config-pptp)#
```

Example 7

Switch to tunnel pppoe 8 configuration mode:

```
esr(config)# tunnel pppoe 8
esr(config-pppoe)#
```

tunnel

The command specifies the encapsulation mode for OPENVPN client.

The use of a negative form (no) of the command sets the default encapsulation.

Syntax

```
tunnel <MODE>
```

Parameters

<MODE> – OPENVPN client encapsulation mode:

- ip – encapsulation of IP packets in OPENVPN;
- ethernet – encapsulation of Ethernet frames in OPENVPN.

Default value

ip

Required privilege level

10

Command mode

CONFIG-OPENVPN

Example

```
esr(config-openvpn)# mode ethernet
```

tunnel-source

This command specifies the VRF name from the IP-interface of which this GRE tunnel will be built. This command is relevant if the GRE-tunnel is built through a different VRF than the VRF of the tunnel itself.

The use of the negative form of the command (no) sets the mode when the GRE tunnel and the IP interface from which the GRE-tunnel is built are in one VRF.

Syntax

```
tunnel-source [ vrf <VRF> ]
[no] tunnel-source
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters.

Without specifying the «vrf» key and the VRF instance name, the global configuration IP interface will be used.

Default value

Disabled (GRE tunnel and IP interface belong to the same VRF).

Required privilege level

10

Command mode

CONFIG-GRE

Example

```
esr(config-gre)# tunnel-source vrf magistral
```

username

The command specifies the user and password to connect to L2TP, PPPoE or PPTP server.

The use of a negative form (no) of the command removes a specified user.

Syntax

```
username <NAME> password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no username <NAME>
```

Parameters

<NAME> – user name, set by the string of up to 31 characters;

<CLEAR-TEXT> – password, set by the string of 1 to 64 characters;

<ENCRYPTED-TEXT> – encrypted password, set by the string of [2..128] characters.

Required privilege level

15

Command mode

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

Example

```
esr(config-pptp)# username fedor password ascii-text password
```

username

This command creates the user to connect to the OPENVPN server.

The use of a negative form (no) of the command removes a specified user.

Syntax

```
username <NAME>

no username <NAME>
```

Parameters

<NAME> – user name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-OPENVPN

Example

```
esr(config-openvpn)# username fedor
```

14 L2 functions management

- L2 functions management
  - bridge
  - bridge-group
  - description
  - enable
  - mac-address
  - ports vrrp filtering
  - protected-ports
  - protected-ports exclude vlan
  - show interfaces bridge
  - thresholds wifi-tunnels-number high
  - thresholds wifi-tunnels-number low
  - unknown-unicast-forwarding disable
  - vlan
- Spanning Tree management
  - instance
  - name
  - revision
  - show spanning-tree
  - show spanning-tree active
  - show spanning-tree bpdu
  - spanning-tree
  - spanning-tree bpdu
  - spanning-tree cost
  - spanning-tree disable
  - spanning-tree forward-time
  - spanning-tree hello-time
  - spanning-tree link-type
  - spanning-tree max-age
  - spanning-tree mode
  - spanning-tree mst
  - spanning-tree mst configuration
  - spanning-tree mst cost
  - spanning-tree mst max-hops
  - spanning-tree mst port-priority
  - spanning-tree pathcost method
  - spanning-tree portfast
  - spanning-tree port-priority
  - spanning-tree priority
- VLAN monitoring and configuration
  - force-up
  - ip internal-usage-vlan
  - name
  - show interfaces switch-port vlans
  - show vlans
  - show vlans internal-usage
  - switchport access vlan
  - switchport default-vlan tagged
  - switchport forbidden default-vlan
  - switchport general acceptable-frame-type
  - switchport general allowed vlan
  - switchport general allowed vlan auto-all

    - switchport general ingress-filtering disable
    - switchport general macs-group
    - switchport general pvid
    - switchport macs-group
    - switchport trunk allowed vlan
    - switchport trunk allowed vlan auto-all
    - switchport trunk native-vlan
    - vlan
- Voice-VLAN monitoring and configuration
    - application
    - dscp
    - lldp network-policy
    - network-policy
    - priority
    - vlan

L2 functions management

bridge

The command adds a network bridge to the system and switches to its configuration mode.

The use of a negative form (no) of the command removes the bridge.

Syntax

```
[no] bridge <BRIDGE-ID>
```

Parameters

<BRIDGE-ID> – bridge ID number, specified in the form described in Section Types and naming order of router interfaces.

Required privilege level

10

Command mode

CONFIG

Example

Switch to network *bridge 10* configuration mode:

```
esr(config)# bridge 10
esr(config-bridge)#
```

bridge-group

The command adds a current network interface to L2 domain.

The use of a negative form (no) of the command removes an interface from L2 domain.

Syntax

```
bridge-group <BRIDGE-ID> [tagged]
no bridge-group
```

Parameters

<BRIDGE-ID> – bridge ID number, specified in the form described in Section Types and naming order of router interfaces.

tagged – key to direct tagged frames of a physical interface to a bridge. Applicable only on the physical interfaces of ESR-10/12V/12VF/14VF/20/21/100/200/3100 routers.

Required privilege level

10

Command mode

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-PORT-CHANNEL

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-L2TPV3

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-OPENVPN

Example

```
esr(config-subif)# bridge-group 15
```

description

The command assigns the description of configured network bridge.

The use of a negative form (no) of the command removes a specified description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – network bridge description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-BRIDGE

Example

```
esr(config-bridge)# description "broadway"
```

enable

This command allows network bridge operation. The network bridge does not work without this command

The use of a negative form (no) of the command disables the data routing.

Syntax

`[no] enable`

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-BRIDGE

Example

```
esr(config-bridge)# enable
```

mac-address

The command specifies the MAC address of network bridge, physical or aggregated interface different from a system one.

The use of a negative form (no) of the command sets the default value.

Syntax

```
mac-address <ADDR>
```

```
no mac-address
```

Parameters

<ADDR> – network bridge MAC address, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF].

Default value

System MAC address.

Required privilege level

15

Command mode

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-bridge)# mac-address A8:F9:B0:00:00:04
```

ports vrrp filtering

The command prohibits the sending of VRRP messages in interfaces included in L2 domain.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] ports vrrp filtering {enable | exclude vlan}
```

Parameters

enable – enabling a prohibition to send VRRP messages in interfaces included in L2 domain;

exclude vlan – when specifying this parameter, VLAN is excluded from the filtration list.

Default value

VRRP messages filtration is disabled.

Required privilege level

10

Command mode

CONFIG-BRIDGE

Example

```
esr(config-bridge)# ports vrrp filtering enable
```

protected-ports

This command manages the isolation mode of the interfaces.

The use of a negative form (no) of the command sets the default value.

Syntax

```
protected-ports <MODE>
```

Parameters

<MODE> – interface isolation mode, may take the following values:

- none – interface isolation disabled. In this mode, frame switching between network bridge members is allowed.
- local – interface isolation is enabled. In this mode, frame switching between network bridge members is prohibited.
- radius – interface isolation is enabled. To use this mode, the WiFi tunnel controller must be set to «radius» mode. In this mode, frame switching between network bridge members is prohibited, except for SoftGRE DATA tunnels. In this mode, frame switching between network bridge members is prohibited, except for SoftGRE DATA

Default value

none

Required privilege level

10

Command mode

CONFIG-BRIDGE

Example

```
esr(config-bridge)# protected-ports local
```

protected-ports exclude vlan

This command removes the VLAN assigned to the network bridge from the list of isolated interfaces in the bridge. This option is relevant when setting protected-ports mode other than the default value.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] protected-ports exclude vlan
```

Parameters

None.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-BRIDGE

Example

```
esr(config-bridge)# protected-ports exclude vlan
```

show interfaces bridge

The command displays information on VLAN, interfaces and tunnels connected by the bridge.

Syntax

```
show interfaces bridge [<BRIDGE-ID>]
```

Parameters

<BRIDGE-ID> – bridge ID number, specified in the form described in Section Types and naming order of router interfaces.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show interfaces bridge 1
Bridges      Interfaces
----------   ----------------------------------------
bridge 1     vlan 1,gi1/0/1.10
```

thresholds wifi-tunnels-number high

⚠ This functionality is supported only if you have a license for WiFi-controller

The command sets the upper alarm threshold in case of exceeding the number of softgre sub tunnels included in the bridge.

The use of a negative form (no) of the command sets the default value.

Syntax

```
thresholds wifi-tunnels-number high<NUM>
no thresholds wifi-tunnels-number high
```

Parameters

<NUM> – upper alarm threshold when exceeding the number of softgre sub tunnels included in the bridge, takes the value in the range of [1..1000]

Default value

950

Required privilege level

10

Command mode

CONFIG-BRIDGE

Example

```
esr(config)# thresholds wifi-tunnels-number high 200
```

thresholds wifi-tunnels-number low

> ⚠ This functionality is supported only if you have a license for WiFi-controller

The command sets the lower alarm threshold in case of exceeding the number of softgre sub tunnels included in the bridge.

The use of a negative form (no) of the command sets the default value.

Syntax

```
thresholds wifi-tunnels-number low <NUM>
no thresholds wifi-tunnels-number low
```

Parameters

<NUM> – lower alarm threshold when exceeding the number of softgre sub tunnels included in the bridge, takes the value in the range of [1..1000]

Default value

925

Required privilege level

10

Command mode

CONFIG-BRIDGE

Example

```
esr(config)# thresholds wifi-tunnels-number low 20
```

unknown-unicast-forwarding disable

The command prohibits the switching of unicast traffic with missing MAC addresses in the network bridge MAC table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] unknown-unicast-forwarding disable
```

Parameters

The command does not contain parameters.

Default value

Unicast traffic switching with unknown MAC addresses is allowed.

Required privilege level

10

Command mode

CONFIG-BRIDGE

Example

```
esr(config-bridge)# unknown-unicast-forwarding disable
```

vlan

The command binds a current network bridge with VLAN. All ports that are members of the assigned VLAN are automatically included in the network bridge and become members of the shared L2 domain. To manage the network interface membership in VLAN, use the commands described in Section VLAN configuration and monitoring.

The use of a negative form (no) of the command removes VLAN binding and disconnects corresponding interfaces from a network bridge.

Syntax

```
vlan <VID>
```

```
no vlan
```

Parameters

<VID> – VLAN identifier, set in the range of [1..4095].

Required privilege level

10

Command mode

CONFIG-BRIDGE

Example

```
esr(config-vlan)# vlan 40
```

Spanning Tree management

⚠ In the current firmware version, this functionality is supported only by ESR-1000 router.

instance

The command generates the match between MSTP instance and VLAN groups.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] instance <INSTANCE> vlan <VID>
```

Parameters

<INSTANCE> – MST instance ID, set in the range of [1..15];

<VID> – VLAN ID, set in the range of [1..4094].

Required privilege level

10

Command mode

CONFIG-MST

Example

```
esr(config-mst)#instance 5 vlan 10-250
```

name

The command specifies MSTP configuration name.

The use of a negative form (no) of the command sets the default value.

Syntax

```
name <NAME>
no name
```

Parameters

<NAME> – MSTP configuration name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-MST

Example

```
esr(config-mst)# name test
```

revision

The command specifies MSTP configuration revision.

The use of a negative form (no) of the command sets the default value.

Syntax

```
revision <NUM>
no revision
```

Parameters

<NUM> – MSTP configuration revision, set in the range of [0..65535].

Default value

0

Required privilege level

10

Command mode

CONFIG-MST

Example

```
esr(config-mst)#revision 5000
```

show spanning-tree

The command displays the detailed information on STP configuration for selected interface or the whole device.

Syntax

```
show spanning-tree { <IF> | bridge }
```

Parameters

<IF> – physical or aggregated interface or a group of physical interfaces is specified in the form described in Section Types and naming order of router interfaces;

bridge – command for displaying the device general information.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show spanning-tree gigabitethernet 1/0/10
Port gi1/0/10 disabled
State: BLK
Port id: ---
Type: ---
Designated bridge Priority: ---
Designated port id: ---
Role: ---
Port cost: ---
Designated path cost: ---
Address: ---
Port Fast: ---
esr# show spanning-tree bridge
Protocol version: STP
     Root ID: [32768] 02:01:02:03:04:55
               Root port: [128] gigabitethernet 1/0/14
               Pathcost 4
               Message Age 1
               Hello time: 2 Max age time: 20 Forward delay: 15
   Bridge ID: [32768] 02:20:03:A0:04:90
               Hello time: 2 Max age time: 20 Forward delay: 15
               Transmit hold count: 6 Topology change: 0
               Time since topology change: 13736 Topology change count: 2 show
```

show spanning-tree active

The command displays information on STP configuration and information on active ports.

Syntax

```
show spanning-tree active
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show spanning-tree active
Protocol version: RSTP
          Root ID: [32768] A8:F9:4B:83:52:C0
                    Root port: [128] gigabitethernet 1/0/20
                    Pathcost 20000
                    Message Age 1
                    Hello time: 2 Max age time: 20 Forward delay: 15
        Bridge ID: [32768] A8:F9:4B:AA:03:00
                    Hello time: 2 Max age time: 20 Forward delay: 15
                    Transmit hold count: 6 Topology change: 0
                    Time since topology change: 2318 Topology change count: 1
Name           State   Prio.Num   Cost        Status     Role       Type
------------   -----   --------   ---------   --------   --------   --------------
gi1/0/20       en      128.2318   20000       FRW        Root       RSTP
```

show spanning-tree bpdu

The command displays the mode of packet processing by BPDU interface.

Syntax

```
show spanning-tree bpdu
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show spanning-tree bpdu
Global: filtering
```

spanning-tree

The command enables Spanning Tree family protocols (STP, RSTP, MSTP) on the router.

The use of a negative form (no) of the command disables Spanning Tree family protocols support.

Syntax

```
[no] spanning-tree
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# spanning-tree
```

spanning-tree bpdu

> ⚠ Unlike other commands in the «Spanning Tree Management» section, this command is available on all ESR router models

The command specifies the mode of packet processing by BPDU interface with disabled STP.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree bpdu <MODE>
no spanning-tree bpdu
```

Parameters

<MODE> – operation mode:

- filtering – BPDU packets are filtrated on the interface with disabled STP;
- flooding – untagged BPDU packets are transmitted on the interface with disabled STP, tagged ones are filtrated.

Default value

flooding

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# spanning-tree bpdu filtering
```

spanning-tree cost

The command sets the method of path cost evaluation.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree cost <COST>
no spanning-tree cost
```

Parameters

<COST> – path cost, set in the range of [1..20000000].

Default value

4

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# spanning-tree cost 115
```

spanning-tree disable

The command prohibits STP operation on a configured interface.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] spanning-tree disable
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# spanning-tree disable
```

spanning-tree forward-time

The command sets the time interval spent on listening to and examining states before switching to the 'transmitting' state.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree forward-time <TIME>
```

```
no spanning-tree forward-time
```

Parameters

<TIME> – time in seconds, takes values of [4..30].

Default value

15

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# spanning-tree forward-time 20
```

spanning-tree hello-time

The command sets the time interval between BPDU packets sending.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree hello-time <TIME>
no spanning-tree hello-time
```

Parameters

<TIME> – time in seconds, takes values of [1..10].

Default value

2

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# spanning-tree hello-time 20
```

spanning-tree link-type

The command sets RSTP to transmission state and defines type of connection for selected port – 'point-to-point', 'shared'.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree link-type {point-to-point | shared}
no spanning-tree link-type
```

Parameters

point-to-point – command defines an interface as 'point-to-point';

shared – command defines an interface as 'shared'.

Default value

point-to-point

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# spanning-tree link-type point-to-point
```

spanning-tree max-age

The command sets STP lifetime.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree max-age <TIME>
no spanning-tree max-age
```

Parameters

<TIME> – time in seconds, takes values of [6..40].

Default value

20

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# spanning-tree max-age 35
```

spanning-tree mode

The command selects supported protocol from STP family.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree mode <MODE>
no spanning-tree mode
```

Parameters

<MODE> – STP family protocol:

- STP – IEEE 802.1D Spanning Tree Protocol;
- RSTP – IEEE 802.1W Rapid Spanning Tree Protocol;
- MSTP – IEEE 802.1s Multiple Spanning Trees.

Default value

RSTP

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# spanning-tree mode STP
```

spanning-tree mst

The command configures the priority for a certain MSTP instance.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree mst <INSTANCE> priority <PRIORITY>
no spanning-tree mst <INSTANCE> priority
```

Parameters

<INSTANCE> – MST instance ID, set in the range of [1..15];

<PRIORITY> – priority, set in the range of [0..61440] in increments of 4096.

Default value

32768

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# spanning-tree mst 14 priority 4096
```

spanning-tree mst configuration

The command switches to MSTP parameters configuration mode.

Syntax

```
spanning-tree mst configuration
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# spanning-tree mst configuration
esr(config-mst)#
```

spanning-tree mst cost

The command sets the method of path cost evaluation for MST instance.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree mst <INSTANCE> cost <COST>
no spanning-tree mst <INSTANCE>
```

Parameters

<INSTANCE> – MST instance ID, set in the range of [1..15];

<COST> – path cost, set in the range of [1..20000000].

Default value

4

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# spanning-tree mst 1 cost 115
```

spanning-tree mst max-hops

The command sets the maximum amount of hops for BPDU packet that are required to build a tree and to keep its structure information. If the packet has already passed the maximum amount of hops, it is dropped on the next hop.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree mst max-hops <NUM>
no spanning-tree mst max-hops
```

Parameters

<NUM> – amount of hops, set in the range of [6..40].

Default value

20

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# spanning-tree mst max-hops 10
```

spanning-tree mst port-priority

The command sets the interface priority for MST instance.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree mst <INSTANCE> port-priority <PRIORITY>
no spanning-tree mst <INSTANCE> port-priority
```

Parameters

<INSTANCE> – MST instance ID, set in the range of [1..15];

<PRIORITY> – priority, set in the range of [0..240] in increments of 16.

Default value

128

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# spanning-tree port-priority 160
```

spanning-tree pathcost method

The command sets the method of path cost evaluation.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree pathcost method {short | long}
no spanning-tree pathcost method
```

Parameters

long – cost value in the range of [1..200000000];

short – cost value in the range of [1..65535].

Default value

short

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# spanning-tree pathcost method short
```

spanning-tree portfast

The command enables the mode in which the port, when the link is brought up, immediately switches to the transmission state without waiting for the timer to expire.

The use of a negative form (no) of the command disables the mode of immediate switch to the transmission state after the link is brought up.

Syntax

```
[no] spanning-tree portfast
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# spanning-tree portfast
```

spanning-tree port-priority

The command sets interface priority in spanning tree.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree port-priority <PRIORITY>
no spanning-tree port-priority
```

Parameters

<PRIORITY> – priority, set in the range of [0..240] in increments of 16.

Default value

128

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# spanning-tree port-priority 160
```

spanning-tree priority

The command configures spanning tree priority.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spanning-tree priority <PRIORITY>
no spanning-tree priority
```

Parameters

<PRIORITY> – priority, set in the range of [0..61440] in increments of 4096.

Default value

32768

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# spanning-tree priority 4096
```

VLAN monitoring and configuration

force-up

The command enables VLAN activity mode regardless of the status of interfaces on which the VLAN’s frames processing is allowed.

The use of a negative form (no) of the command disables VLAN activity mode.

Syntax

```
[no] force-up
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-VLAN

Example

```
esr(config-vlan)# force-up
```

ip internal-usage-vlan

> ⚠ In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

The command is used for VLAN redundancy for internal use on the interface.

The use of a negative form (no) of the command cancels the redundancy.

Syntax

```
ip internal-usage-vlan <VID>
no ip internal-usage-vlan
```

Parameters

<VID> – VLAN ID, set in the range of [2..4094].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# ip internal-usage-vlan 1500
```

name

The command is used to add a VLAN description.

The use of a negative form (no) of the command sets the default value.

Syntax

`name <NAME>`

`no name`

Parameters

<NAME> – VLAN description, set by the string of up to 255 characters.

Default value

The description is not specified.

Required privilege level

10

Command mode

CONFIG-VLAN

Example

```
esr(config)# name L2-ACCESS
```

show interfaces switch-port vlans

> ⚠ In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

The command displays VLAN interface participation mode.

Syntax

`show interfaces switch-port vlans [<IF>]`

Parameters

<IF> – a physical or aggregated interface's name specified in the form described in Section Types and naming order of router interfaces.

You may specify several interfaces in the command. If interface indexes are not specified, then the information on all interfaces of a specified group will be shown. When executing a command without parameter, all physical interface information will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show interfaces switch-port vlans gigabitethernet 1/0/1-7
Interface   PVID   Frame types     Ingress     Tagged        Untagged
                                   filtering
---------   ----   -------------   ---------   -----------   -----------
gi1/0/1     1      All             yes         101           1
gi1/0/2     1      All             yes         150-151       1
gi1/0/3     1      All             yes         none          1
gi1/0/4     1      All             yes         none          1
gi1/0/5     1      All             yes         55            1
gi1/0/6     1      All             yes         none          1
gi1/0/7     1      All             yes         none          1
N/A - interface doesn't exist
N/S - interface is not a 802.1Q bridge port
ERR - can't get vlan setting for interface
```

show vlans

The command displays a certain VLAN information.

Syntax

```
show vlans [<VID>]
```

Parameters

<VID> – VLAN ID, takes the values of [1..4094].

You can specify several VLANs separated by commas ',' or you can specify the range of VLANs with '-'. When executing a command without parameter, all created VLANs will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show vlans
VID    Name                    Tagged                   Untagged
----   ---------------------   ----------------------   ----------------------
1      default                                          gi1/0/3-4, gi1/0/6-24,
                                                        po1
2      --                                               gi1/0/1, te1/0/1-2
```

show vlans internal-usage

> ⚠ In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

The command displays the information on VLANs used by the system.

Syntax

```
show vlans internal-usage
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show vlans internal-usage
Usage      VID    Reserved   IP address
--------   ----   --------   ----------
gi1/0/18   4088   No         Active
gi1/0/16   4089   No         Active
gi1/0/15   4090   No         Active
```

switchport access vlan

> ⚠ In the current firmware version this functionality is supported by ESR-10/12V/12VF/14VF/20/21/100/200/3100 routers only

The command includes/removes interface in/from VLAN in access operation mode.

Syntax

```
switchport access vlan <VID>
```

```
no switchport access vlan
```

Parameters

<VID> – VLAN ID, set in the range of [2..4094].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# switchport access vlan 50
```

switchport default-vlan tagged

The command changes the default VLAN interface membership for tagged one.

The use of a negative form (no) of the command changes the default VLAN interface membership for untagged one.

Syntax

```
[no] switchport default-vlan tagged
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# switchport default-vlan tagged
```

switchport forbidden default-vlan

The command removes the interface from default VLAN.

The use of a negative form (no) of the command enables adding vlan on a port.

Syntax

```
[no] switchport forbidden default-vlan
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# no switchport forbidden default-vlan
```

switchport general acceptable-frame-type

⚠ In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

The command sets the type of frames that can be received by the interface.

The use of a negative form (no) of the command sets the default value.

Syntax

```
switchport general acceptable-frame-type { tagged-only | all }
no switchport general acceptable-frame-type
```

Parameters

tagged-only – receive only tagged frames;

all – receive all frames.

Default value

all

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config)# switchport general acceptable-frame-type tagged-only
```

switchport general allowed vlan

⚠ In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

The command includes/removes interface in/from VLAN in access operation mode.

Syntax

```
switchport general allowed vlan <ACT> <VID> [<TYPE>]
```

Parameters

<ACT> – allocated action:

- add – interface addition into VLAN;
- remove – interface removal from VLAN.

<VID> – VLAN ID, set in the range of [2..4094]. You can also specify it by the range with '-' or by comma-separated list;

<TYPE> – packet type:

- tagged – interface will send and receive packets as tagged in specified VLANs;
- untagged – interface will send and receive packets as untagged in specified VLANs. The VLAN to which incoming untagged packets will be sent is configured using the *switchport general pvid* command described in switchport general pvid.

Default value

If you do not specify the <TYPE> parameter, then the default is 'tagged'.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example 1

Exclude interface from VLAN 50 membership

```
esr(config-if-gi)# switchport general allowed vlan remove 50
```

Example 2

Include interface into VLAN 10-5- as tagged

```
esr(config-if-gi)# switchport general allowed vlan add 10-50
```

switchport general allowed vlan auto-all

> ⚠ In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

This command enables the automatic addition of an interface to all VLANs created on the router.

The use of the negative form of the command (no) disables the automatic addition of a port to all VLANs created on the router.

Syntax

```
[no] switchport general allowed vlan auto-all [ <TYPE> ]
```

Parameters

<TYPE> – packet type:

- tagged – interface will send and receive packets as tagged in specified VLANs;

- untagged – interface will send and receive packets as untagged in specified VLANs. The VLAN to which incoming untagged packets will be sent is configured using the switchport general pvid command described in switchport general pvid.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# switchport general allowed vlan auto-all
```

switchport general ingress-filtering disable

> ⚠ In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

The command disables filtration of incoming packets on the basis of VLAN ID value assigned.

The use of a negative form (no) of the command enables the interface being configured.

Syntax

```
switchport general ingress-filtering disable
no switchport general ingress-filtering
```

Parameters

The command does not contain parameters.

Default value

Filtration is enabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# switchport general ingress-filtering disable
```

switchport general macs-group

⚠ In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

This command assigns a MAC address profile and maps the VLAN-ID to which packets will fall with the MAC address of the source that is part of the MAC address profile. Information about configuring profiles can be found in the Profiles management section.

The use of a negative form (no) of the command removes a specified profile of dynamic authorization servers (DAS).

Syntax

`switchport general macs-group <NAME> vlan <VID>`

`no switchport general macs-group <NAME>`

Parameters

<NAME> – MAC addresses profile name, set by the string of up to 31 characters.

<VID> – VLAN ID, set in the range of [2..4094].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# switchport general macs-group OGM1 vlan 999
```

switchport general pvid

> ⚠ In the current firmware version this functionality is supported by ESR-1000/1200/1500/1511/1700 routers only

This command sets the VLAN Port ID (PVID) for incoming untagged traffic.

The use of a negative form (no) of the command sets the default value.

Syntax

```
switchport general pvid <VID>
no switchport general pvid
```

Parameters

<VID> – VLAN ID, set in the range of [1..4094].

Default value

1

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# switchport general pvid 999
```

switchport macs-group

> ⚠ In the current firmware version this functionality is supported by ESR-10/12V/12VF/14VF/20/21/100/200/3100 routers only

This command assigns a MAC address profile and maps the VLAN-ID to which packets will fall with the MAC address of the source that is part of the MAC address profile. Information about configuring profiles can be found in the Profiles management section.

The use of a negative form (no) of the command removes the assignment of MAC addresses and VLAN ID.

Syntax

```
switchport macs-group <NAME> vlan <VID>
no switchport macs-group <NAME>
```

Parameters

<NAME> – MAC addresses profile name, set by the string of up to 31 characters.

<VID> – VLAN ID, set in the range of [2..4094].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# switchport macs-group OGM1 vlan 999
```

switchport trunk allowed vlan

> In the current firmware version this functionality is supported by ESR-10/12V/12VF/14VF/20/21/100/200/3100 routers only

The command includes/removes interface in/from VLAN in trunk operation mode.

Syntax

```
switchport trunk allowed vlan <ACT> <VID>
```

Parameters

<ACT> – allocated action:

- add – interface addition into VLAN;
- remove – interface removal from VLAN;

<VID> – VLAN ID, set in the range of [2..4094]. You can also specify it by the range with '-' or by comma-separated list.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# switchport trunk allowed vlan add 10-50
```

switchport trunk allowed vlan auto-all

> ⚠ In the current firmware version this functionality is supported by ESR-10/12V/12VF/14VF/20/21/100/200/3100 routers only

This command enables the automatic addition of a port to all VLANs created on the router.

The use of the negative form of the command (no) disables the automatic addition of a port to all VLANs created on the router.

Syntax

```
[no] switchport trunk allowed vlan auto-all
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-IF-GI

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# switchport trunk allowed vlan auto-all
```

switchport trunk native-vlan

> In the current firmware version this functionality is supported by ESR-10/12V/12VF/14VF/20/21/100/200/3100 routers only

The command adds/removes interface in/from VLAN in trunk operation mode. All untagged traffic arriving at this interface is routed to this VLAN.

Syntax

```
switchport trunk native-vlan <VID>
no switchport trunk native-vlan
```

Parameters

<VID> – VLAN ID, set in the range of [2..4094].

Default value

1

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# switchport trunk native-vlan 55
```

vlan

The command adds a VLAN to the system and switches to its configuration mode. There is always a VLAN with ID 1 on the router, all interfaces are added by default in this VLAN.

The use of a negative form (no) of the command removes the VLAN.

Syntax

```
[no] vlan <VID>
```

Parameters

<VID> – VLAN ID, set in the range of [2..4094].

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# vlan 40
```

Voice-VLAN monitoring and configuration

application

This command configures the application that will be listed in the LLDP-MED packet when sending LLDPDU from interfaces on which network-policy is assigned.

The use of a negative form (no) of the command removes domain name for the router.

Syntax

```
application <APP-TYPE>
```

```
no application
```

Parameters

<APP-TYPE> – type of the application for which network-policy will be enabled. Takes the following values:

- voice;
- voice-signaling;
- guest-voice;
- guest-voice-signaling;
- softphone-voice;
- video-conferencing;
- streaming-video;
- video-signaling.

Default value

None

Required privilege level

10

Command mode

CONFIG-NET-POLICY

Example

```
esr(config-net-policy)# application voice
```

dscp

This command configures DSCP value that will be listed in the LLDP-MED packet when sending LLDPDU from interfaces on which network-policy is assigned.

The use of negative form of the command (no) removes the DSCP distribution for a specified network policy (network-policy).

Syntax

`dscp <DSCP>`

`dscp`

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Default value

DSCP sending in LLDPDU-MED is disabled.

Required privilege level

10

Command mode

CONFIG-NET-POLICY

Example

```
esr(config-net-policy)# dscp 62
```

lldp network-policy

This command assigns the network policy that created earlier to the interface (network-policy).

The use of negative form of the command (no) removes the previously assigned network policy from the interface.

> ⚠ LLDPDU messages described by the policy will be sent only if LLDP-MED support is enabled in the global configuration (see the command in section lldp med fast-start enable).

Syntax

```
[no] lldp network-policy <NAME>
```

Parameters

<NAME> – name of the policy being assigned, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example

```
esr(config-if-gi) lldp network-policy ip-phones
```

network-policy

The command creates a network policy to the system and switches to its configuration mode.

The use of negative form of the command (no) removes the previously created network policy.

Syntax

```
[no] network-policy <NAME>
```

Parameters

<NAME> – name of the policy being created, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# network-policy ip-phones
```

priority

This command configures COS value that will be listed in the LLDP-MED packet when sending LLDPDU from interfaces on which network-policy is assigned.

The use of negative form of the command (no) removes the COS distribution for a specified network policy (network-policy).

Syntax

```
priority <COS>
```

```
no priority
```

Parameters

<COS> – priority value, takes the following values:

- best-effort – COS0;
- background – COS1;
- excellent-effort – COS2;
- critical-applications – COS3;
- video – COS4;
- voice – COS5;
- internetwork-control – COS6;
- network-control – COS7.

Default value

COS sending in LLDPDU-MED is disabled.

Required privilege level

10

Command mode

CONFIG-NET-POLICY

Example

```
esr(config-net-policy)# priority voice
```

vlan

This command configures VLAN value that will be listed in the LLDP-MED packet when sending LLDPDU from interfaces on which network-policy is assigned.

The use of negative form of the command (no) removes the VLAN configuration for a specified network policy (network-policy).

Syntax

```
vlan <VID> [tagged]
```

Parameters

<VID> – VLAN ID, takes values of [1..4094];

tagged – key, during the installation of which, the subscriber device will send Ethernet frames of the specified application in a tagged form.

Default value

None

Required privilege level

10

Command mode

CONFIG-NET-POLICY

Example

```
esr(config-net-policy)# vlan 3258 tagged
```

15 Address table operation

- clear arp-cache
- clear ipv6 neighbors
- clear mac address-table
- ip arp
- ip arp reachable-time
- ipv6 nd
- ipv6 nd reachable-time
- mac address-table aging-time
- mac address-table save-secure-freq
- port-security max
- port-security mode
- port-security unknown-sa-action
- show arp
- show arp configuration
- show ipv6 neighbors
- show ipv6 neighbors configuration
- show mac address-table

clear arp-cache

The commands clears ARP table.

Syntax

```
clear arp-cache [ <OPTIONS> ]
```

Parameters

<OPTIONS> – command parameters for detailed information, optional parameter:

- <VRF> – VRF instance name, set by the string of up to 31 characters. An optional parameter that, if specified, will clear the ARP table in the specified VRF;
- <IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;
- <ADDR> – default gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];
- <ADDR> – MAC address to be searched, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF].

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear arp-cache ip-address 10.0.0.8
```

clear ipv6 neighbors

The command clears the IPv6 Neighbor Discovery tables contents.

Syntax

```
clear ipv6 neighbors [<OPTIONS> ]
```

Parameters

<OPTIONS> – command parameters for detailed information, optional parameter:

- <VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, IPv6 Neighbor Discovery table will be cleared in a specified VRF;
- <IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;
- ipv6-address <IPV6-ADDR> – IPv6 address to be searched, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].
- <ADDR> – MAC address to be searched, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF].

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear ipv6 neighbors
```

clear mac address-table

The command is used to delete information about learned MAC addresses.

Syntax

```
clear mac address-table [ { dynamic | static } ] [ { interface { <IF> | host-port <U/S/
P> } | bridge <BRIDGE-ID> | tunnel { gre | softgre } <ID> | vlan <VLAN-ID> }
```

Parameters

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;

<U/S/P> – Unit (1), slot (0) and interface number of the packet processor;

<BRIDGE-ID> – bridge ID number, specified in the form described in Section Types and naming order of router interfaces;

<ID> – tunnel identifier;

<VLAN ID> – VLAN number. It is possible to specify a vlan list using the ',' symbol without spaces, a vlan range using the '-' symbol and/or a combination of lists and ranges.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear mac address-table
```

ip arp

This command adds a static entry to the ARP table.

The use of a negative form (no) of the command removes the static entry from the ARP table.

Syntax

```
ip arp [ vrf <VRF> ] <IP> <MAC> { <IF> | <TUN> }
no ip arp [ vrf <VRF> ] <IP>
```

Parameters

<VRF> – VRF name, set by the string of up to 31 characters.

<IP> – host IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<MAC> – MAC address of the client, which will be given the IP address, defined as XX: XX: XX: XX: XX: XX where each part takes the values of [00..FF];

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config-if-gi)# ip arp 192.168.54.22 a8:f9:4b:ab:2e:d0 bridge 3
```

ip arp reachable-time

The command sets lifetime of the record in the ARP table.

The use of a negative form (no) of the command sets the default value of arp reachable-time parameter.

Syntax

```
ip arp reachable-time <TIME>
no ip arp reachable-time
```

Parameters

<TIME> – lifetime of dynamic MAC addresses, in milliseconds. Allowed values are from 5000 to 100000000 milliseconds. Real time of the entry update varies from [0,5;1,5]*<TIME>.

Required privilege level

10

Default value

160000

Command mode

CONFIG

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip arp reachable-time 6000
```

ipv6 nd

This command adds a static entry to the ND table.

The use of a negative form (no) of the command removes the entry from the ND table.

Syntax

```
ipv6 nd [ vrf <VRF> ] <IPV6> <MAC> {<IF> | <TUN>}
```

```
no ipv6 nd [ vrf <VRF> ] <IP>
```

Parameters

<VRF> – VRF name, set by the string of up to 31 characters.

<IPV6-ADDR> – IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

<MAC> – MAC address of the client, which will be given the IP address, defined as XX: XX: XX: XX: XX: XX where each part takes the values of [00..FF];

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config-if-gi)# ip arp 192.168.54.22 a8:f9:4b:ab:2e:d0 bridge 3
```

ipv6 nd reachable-time

This command sets the time during which the remote IPv6 host is considered available when there is no host activity.

The use of a negative form (no) of the command sets the default value of nd reachable-time parameter.

Syntax

```
ipv6 nd reachable-time <TIME>
no ipv6 nd arp reachable-time
```

Parameters

<TIME> is the lifetime of an IPv6 remote node entry in the ND protocol table, in milliseconds. Allowed values are from 5000 to 100000000 milliseconds. Real time of the entry update varies from [0,5;1,5]*<TIME>.

Default value

30000

Required privilege level

10

Command mode

CONFIG

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-LOOPBACK

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 nd reachable-time 27000
```

mac address-table aging-time

The command sets the lifetime of dynamic MAC addresses in forwarding table.

The use of a negative form (no) of the command sets the default 'aging time'.

Syntax

```
mac address-table aging-time <AGING TIME>
[no] mac address-table aging time
```

Parameters

<AGING TIME> – lifetime of dynamic MAC addresses, in seconds. Allowed values:

- ESR-1000/1200/1500/1511/1700 – from 10 to 630 seconds. If set to 0, the timer is off.
- ESR-10/12V/12VF/14VF/20/21/100/200 – from 20 to 630 seconds.

Default value

300

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# mac address-table aging-time 30
```

mac address-table save-secure-freq

⚠ In the current firmware version, the command is supported only by ESR-1000 routers

The command sets the frequency of saving secure MAC addresses list.

The use of a negative form (no) of the command sets the default 'mac address-table save-secure-freq' value.

Syntax

```
mac address-table save-secure-freq <SAVE-SECURE-FREQ>
[no] mac address-table save-secure-freq
```

Parameters

<SAVE-SECURE-FREQ> – frequency of saving secure MAC addresses list, takes the value of [600..86400] seconds.

Default value

1200 seconds

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# mac address-table save-secure-freq 650
```

port-security max

⚠ In the current firmware version, this functionality is supported only by ESR-1000 router.

The command sets the maximum number of MAC addresses allowed to be stored on port.

The use of a negative form (no) of the command disables 'port-security'.

Syntax

```
port-security max <MAX>
```

```
no port-security max
```

Parameters

<MAX> – maximum amount of MAC addresses to be stored by port, takes the values of [1..1024].

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

Example

```
esr(config-if-gi)# port-security max 1
```

port-security mode

> ⚠ In the current firmware version, this functionality is supported only by ESR-1000 router.

The command configures 'port-security' mode.

The use of a negative form (no) of the command disables the security mode.

Syntax

```
port-security mode [<OPTIONS>]
```

```
no port-security mode
```

Parameters

<OPTIONS> – parameters of command to select 'port-security' mode:

limited – when enabling the mode:

- all learned MAC addresses are removed from a port;
- amount of addresses that port can store is limited by the current configuration;
- MAC addresses are not saved between hard resets;
- MAC addresses storage time depends on the lifetime of dynamic MAC addresses in forwarding table.

lock – when enabling the mode:

- all learned MAC addresses are saved on a port;
- port does not store new addresses;
- MAC addresses are saved between hard resets;
- MAC addresses storage time depends on the lifetime of dynamic MAC addresses in forwarding table.

secure-delete-on-reset – when enabling the mode:

- all learned MAC addresses are removed from a port;
- amount of addresses that port can store is limited by the current configuration;
- MAC addresses are not saved between hard resets;
- MAC addresses storage time does not depend on the lifetime of dynamic MAC addresses in forwarding table.

secure-permanent – when enabling the mode:

- all learned MAC addresses are removed from a port;
- amount of addresses that port can store is limited by the current configuration;
- MAC addresses are saved between hard resets;
- MAC addresses storage time does not depend on the lifetime of dynamic MAC addresses in forwarding table.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

Example

```
esr(config-if-gi)# port-security mode secure-delete-on-reset
esr(config-if-gi)# port-security mode secure-permanent
```

port-security unknown-sa-action

⚠ In the current firmware version, this functionality is supported only by ESR-1000 router.

The command prohibits the transmission of packets with unknown MAC addresses.

The use of a negative form (no) of the command enables the transmission of packets with unknown MAC addresses.

Syntax

`port-security unknown-sa-action discard`

`no port-security unknown-sa-action`

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

Example

```
esr(config-if-gi)# port-security unknown-sa-action discard
```

show arp

The commands displays ARP table.

Syntax

`show arp [<OPTIONS>]`

Parameters

<options> – command parameters for detailed information, optional parameter:

- <VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, ARP table will be displayed in a specified VRF;
- <IF> – name of an interface or a list of interfaces is specified in the form described in Section Types and naming order of router interfaces. Only information on specified interfaces is displayed;
- <TUN> – names of tunnels are specified as described in section Types and naming order of router tunnels.
- mac-address <MAC> – MAC address to be searched, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF];
- ip-address <ADDR> – IP address to be searched, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

1

Command mode

ROOT

Example

```
esr# show arp
Interface         IP address        MAC address         State             Age(min)
---------------   ---------------   -----------------   ---------------   ----------
bridge 1          192.168.1.1       a8:f9:4b:aa:00:40   --                --
gi1/0/5           10.255.100.1      d8:50:e6:d2:f0:46   reachable         2
gi1/0/5           10.255.100.5      a8:f9:4b:aa:00:45   --                --
```

show arp configuration

The command displays the values of ARP table entries lifetime.

Syntax

```
show arp configuration <IF>
```

Parameters

<IF> – system interface names, specified in the form described in Section Types and naming order of router interfaces;

Required privilege level

1

Command mode

ROOT

Example

```
esr# sh arp configuration gigabitethernet 1/0/1-5
Globally configured ARP reachable time is 6000 msec
Interface          ARP reachable time, msec
---------------    -------------------------
gi1/0/1            6000
gi1/0/2            6000
gi1/0/3            6000
gi1/0/4            6000
gi1/0/4            6000
```

show ipv6 neighbors

The command displays IPv6 Neighbor Discovery tables.

Syntax

```
show ipv6 neighbors [<OPTIONS>]
```

Parameters

<OPTIONS> – command parameters for detailed information, optional parameter:

- <VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, IPv6 Neighbor Discovery table will be displayed in a specified VRF;
- <IF> – name of an interface or a list of interfaces is specified in the form described in Section Types and naming order of router interfaces. Only information on specified interfaces is displayed;
- mac-address <MAC> – MAC address to be searched, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF];
- ipv6-address <IPV6-ADDR> – IPv6 address to be searched, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 neighbors
Interface         IPv6 address               MAC address         State             Age(min)
---------------   -------------------------  -----------------   ---------------   ----------
gi1/0/5           fc00::1                    d8:50:e6:d2:f0:46   reachable         1
gi1/0/5           fc00::2                    a8:f9:4b:aa:00:45   --                --
bridge 1          fe80::aaf9:4bff:feaa:40    a8:f9:4b:aa:00:40   --                --
bridge 2          fe80::aaf9:4bff:feaa:40    a8:f9:4b:aa:00:40   --                --
gi1/0/5           fe80::aaf9:4bff:feaa:45    a8:f9:4b:aa:00:45   --                --
gi1/0/5           ff02::16                   33:33:00:00:00:16   norarp            --
gi1/0/5           ff02::fb                   33:33:00:00:00:fb   norarp            --
gi1/0/5           ff02::1:ff00:1             33:33:ff:00:00:01   norarp            --
gi1/0/5           ff02::1:ff00:2             33:33:ff:00:00:02   norarp            --
```

show ipv6 neighbors configuration

The command displays the lifetime values of a remote node entry in the ND protocol table.

Syntax

`show ipv6 neighbors configuration <IF>`

Parameters

<IF> – system interface names, specified in the form described in Section Types and naming order of router interfaces.

Required privilege level

1

Command mode

ROOT

Example

```
esr# sh ipv6 neighbors configuration tengigabitethernet 1/0/1-2
Globally configured NDP reachable time is 30000 msec
Interface         ND reachable time, msec
---------------   -------------------------
te1/0/1           30000
te1/0/2           30000
```

show mac address-table

The command is used to view the information in the MAC address table.

Syntax

`show mac address-table [ { count { bridge <BRIDGE-ID> | l2vpn p2p <P2P-NAME> } | { static | dynamic} { bridge <BRIDGE-ID> | l2vpn p2p <P2P-NAME> | interface { <IF> | host-port <U/S/P> } | tunnel { gre | softgre } <ID> | vlan <VLAN-ID> | mac <MAC-ADDR> <MAC-MASK> } } ]`

Parameters

- count – show the number of entries in MAC table. The list of MAC addresses is not displayed;
- static – view entries in the MAC address table entered from the router configuration;
- dynamic – view entries in the MAC address table recorded during the MAC address learning;
- <BRIDGE-ID> – bridge ID number, specified in the form described in Section Types and naming order of router interfaces;
- <P2P-NAME> — name of existent p2p service, set by the string of up to 31 characters;
- <IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;
- <HOST-U/S/P> – packet processor interface. Specified as unit (1), slot (0) and interface number of the packet processor;
- <ID> – the identifier of the tunnel is specified as described in section Types and naming order of router tunnels;
- <VLAN ID> – VLAN number. It is possible to specify a vlan list using the «,» symbol without spaces, a vlan range using the «-» symbol and/or a combination of lists and ranges;
- <MAC-ADDR> – MAC address to be searched, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF];
- [MAC-MASK] – MAC address mask, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF]. Mask bits, set to zero, specify MAC address bits excluded from the comparison when searching. Mask default value: FF:FF:FF:FF:FF:FF.

Required privilege level

10

Command mode

ROOT

Example

```
esr# show mac address-table
VID     MAC Address          Interface                        Type
-----   ------------------   ------------------------------   -------
102     a8:f9:4b:aa:44:bb    host-port 1/0/2                  Dynamic
101     a8:f9:4b:aa:44:bb    host-port 1/0/2                  Dynamic
100     a8:f9:4b:aa:44:bb    host-port 1/0/2                  Dynamic
3 valid mac entries
```

16 VRF Configuration

- description
- ip source-vrf
- ip vrf
- ip vrf forwarding
- show ip vrf

description

The command assigns the description of configured VRF.

The use of a negative form (no) of the command removes a specified description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – VRF instance description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-VRF

Example

```
esr(config-vrf)# description "VRF1"
```

ip source-vrf

This command transfers the copying system by TFTP, SCP, FTP, SFTP to VRF.

The use of a negative form (no) of the command switches copying system to standard mode.

Syntax

```
ip { tftp | ftp | ssh | sftp | http } source-vrf <VRF>
no ip { tftp | ftp | ssh | sftp | http } source-vrf
```

Parameters

<VRF> – VRF name, set by the string of up to 31 characters.

tftp – copy by TFTP;

ftp – copy by FTP;

ssh – copy by SCP;

sftp – copy by SFTP;

http – copy by HTTP.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip tftp source-vrf vrf1
```

ip vrf

The command adds VRF to the system and switches to its configuration mode.

The use of a negative form (no) of the command removes VRF from the system.

Syntax

```
[no] ip vrf <VRF>
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip vrf VRF1
esr(config-vrf)#
```

ip vrf forwarding

Specify the name of VRF instance in which the specified network interface, bridge, security zone, dynamic authorization server (DAS) or NAT rules group will be used.

The use of negative form of the command (no) removes the binding of the network interface, bridge, security zone, or group of NAT rules to the VRF instance.

Syntax

```
ip vrf forwarding <VRF>
no ip vrf forwarding
```

Parameters

<VRF> – VRF name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-GRE

CONFIG-LT

CONFIG-SUBTUNNEL

CONFIG-ZONE

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

CONFIG-DAS-SERVER

CONFIG-SNMP-USER

CONFIG-SNMP-VIEW

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-OPENVPN

CONFIG-IPSEC-VPN

Example

```
esr(config-snat-ruleset)# ip vrf forwarding VRF1
```

show ip vrf

The command displays the information on VRF instances existing in the system.

Syntax

show ip vrf [<VRF>]

Parameters

<VRF> – VRF name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip vrf
Name                              Interfaces
-------------------------------   ------------------------------------------
VRF1                              gi1/0/8
VRF2                              gi1/0/10.22
test                              gi1/0/2
```

17 IP addressing configuration

- ip address
- ip redirects
- ip route source-route
- ip unnumbered
- ip unreachables
- show ip interfaces

ip address

The command creates IP interface and adds IP address and subnet mask for a configured interface: physical interface, link aggregation group, tunnel or network bridge.

The use of a negative form (no) of the command removes IP address from the interface. When removing the last address, IP interface is destroyed.

> ⚠ When creating IP interface, the system reserves the biggest vacant VLAN ID that will be used within the system. VLAN is reserved for each IP interface on Ethernet port. If VLAN has already been reserved for IPv6 interface, then VLAN will not be reserved for IP interface.
> You may reserve VLAN ID for internal use by ip internal-usage vlan <VLAN_ID> command.

Syntax

```
ip address <ADDR/LEN>
```

```
no ip address { <ADDR/LEN> | all }
```

Parameters

<ADDR/LEN> – IP address and subnet mask length, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32]. You can specify several IP addresses separated by commas. Up to 8 IP addresses (including IPv6 addresses) can be assigned to the interface. The use of a negative form (no) of the command with 'all' parameter removes all IP addresses.

all – the command deletes all IP addresses on interface.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-E1

CONFIG-MULTILINK

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip address 192.168.25.25/24
```

ip redirects

The command enables the mechanism of sending ICMP messages about the existence of a router with a higher priority in the IP network for a specific IP destination.

The use of a negative form (no) of the command disables the mechanism of sending ICMP messages about the existence of a router with a higher priority in the IP network for a specific IP destination.

Syntax

```
[no] ip redirects
```

Parameters

The command does not contain parameters.

Required privilege level

15

Default value

Sending of ICMP messages about the existence of a router with a higher priority in the IP network for a specific IP destination is enabled.

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-E1

CONFIG-MULTILINK

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Example

```
esr(config-if-gi)# no ip redirects
```

ip route source-route

This command enables the ip source-route option on the router.

The use of a negative form of the command (no) disables the support for the ip source-route option.

Syntax

```
[no] ip route source-route
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip route source-route
```

ip unnumbered

This command enables the interface operation mode with the use of the ip address assigned to another interface.

The use of a negative form (no) of the command disables the ip unnumbered mode.

Syntax

```
ip unnumbered { <IF> | <TUN> }
```

```
no ip unnumbere
```

Parameters

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – names of tunnels are specified as described in section Types and naming order of router tunnels.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-E1

CONFIG-MULTILINK

CONFIG-GRE

CONFIG-IP4IP4

Example

```
esr(config-e1)# ip unnumbered gigabitethernet 1/0/1
esr(config-e1)#
```

ip unreachables

The command enables the sending of ICMP packets about destination address unavailability.

The use of a negative form (no) of the command disables the sending of ICMP packets about destination address unavailability.

Syntax

```
[no] ip unreachables
```

Parameters

The command does not contain parameters.

Required privilege level

15

Default value

Sending of ICMP packets about destination address unavailability.

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-E1

CONFIG-MULTILINK

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Example

```
esr(config-if-gi)# no ip unreachables
```

show ip interfaces

The command displays the information on IP interfaces existing in the system.

Syntax

```
show ip interfaces [ { <IF> | <TUN> | vrf <VRF> [ ip-address <ADDR> ] } ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, IP interfaces will be displayed in a specified VRF;

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. When specifying this parameter, IP interface with a specified IP address will be displayed;

<IF> – system interface names, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – names of tunnels are specified as described in section Types and naming order of router tunnels.

You may specify several system interfaces in the command. If you do not specify interface indexes, then all IP interfaces related to the system interfaces of the specified type will be displayed.

If the command specifies a certain system interface that receives IP parameters via DHCP, DHCP client settings and the status of IP parameters current lease will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip interfaces
IP address             Interface        Type
-------------------    -------------    -------
155.0.0.60/24          br10             static
16.0.0.2/24            gi1/0/16         DHCP
10.0.0.1/8             gi1/0/18         static
180.0.0.1/24           gi1/0/18         static
192.168.1.1/24         gi1/0/2          static
25.0.0.2/30            gre 25           static
10.1.0.2/24            ip4ip4 10        static
esr# show ip interfaces gigabitethernet 1/0/16
IP address             Interface        Type
-------------------    -------------    -------
16.0.0.2/24            gi1/0/16         DHCP
 DHCP Client settings:
 DHCP Server:           N/A
 Lease time(dd:hh:mm): 00:02:00
 Reboot time:           10 seconds
 Retry time:            300 seconds
 Timeout:               60 seconds
 Select timeout:        0 seconds
 Vendor class ID:       N/A
 Ignore options:
    router

 Latest lease contents:
 Lease time(dd:hh:mm): 00:02:00
 DHCP message type:     DHCPACK
 Renew at:              Wednesday2015/02/25 12:22:24 2015/02/25 12:22:24
 Rebind at:             Wednesday2015/02/25 13:14:09 2015/02/25 13:14:09
 Expires at:            Wednesday2015/02/25 13:29:09 2015/02/25 13:29:09
```

18 IPv6 addressing configuration

- ipv6 address
- ipv6 enable
- ipv6 nd managed-config-flag
- ipv6 nd ns-interval
- ipv6 nd other-config-flag
- ipv6 nd prefix
- ipv6 nd ra hop-limit
- ipv6 nd ra lifetime
- ipv6 nd ra max-interval
- ipv6 nd ra min-interval
- ipv6 nd router-preference
- ipv6 redirects
- ipv6 route source-route
- ipv6 unreachables
- show ipv6 interfaces

ipv6 address

The command creates IPv6 interface and adds IPv6 address and subnet mask for a configured interface: physical interface, link aggregation group, tunnel or network bridge.

The use of a negative form (no) of the command removes IPv6 address from the interface. When removing the last address, IPv6 interface is destroyed.

> ⚠ When creating IPv6 interface, the system reserves the biggest vacant VLAN ID that will be used within the system. VLAN is reserved for each IPv6 interface on Ethernet port. If VLAN has already been reserved for IP interface, then VLAN will not be reserved for IPv6 interface.
> You may reserve VLAN ID for internal use by ip internal-usage vlan <VLAN_ID> command.

Syntax

```
ipv6 address <IPV6-ADDR/LEN>
```

```
no ipv6 address {<IPV6-ADDR/LEN>|all}
```

Parameters

<IPV6-ADDR/LEN> – IP address and prefix of a subnet, defined as X:X:X:X::X/EE where each X part takes values in hexadecimal format [0..FFFF] and EE takes values of [1..128]. You can specify several IPv6 addresses separated by commas. Up to 8 IPv6 addresses (including IP addresses) can be assigned to the interface. The use of a negative form (no) of the command with ‘all’ parameter removes all IPv6 address profiles.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 address fc00::1/120
```

ipv6 enable

The command enables IPv6 on the interface.

The use of a negative form (no) of the command disables IPv6 on the interface.

Syntax

```
[no] ipv6 enable
```

Parameters

The command does not contain parameters.

Default value

IPv6 support is disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 enable
```

ipv6 nd managed-config-flag

This command sets managed-config-flag in SLAAC messages. Using this flag determines the receipt of all IPv6 settings from the DHCPv6 server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] ipv6 nd managed-config-flag
```

Parameters

None.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-LOOPBACK

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config)# ipv6 nd managed-config-flag
```

ipv6 nd ns-interval

This command sets the interval for sending IPv6 neighbor search solicitation messages for a broadcast domain.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 nd ns-interval <TIME>
no ipv6 nd ns-interval
```

Parameters

<TIME> – transmission interval, specifies in milliseconds [1..172800000].

Default value

0

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-LOOPBACK

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config)# ipv6 nd ns-interval 2000
```

ipv6 nd other-config-flag

This command sets other-config-flag in SLAAC messages. The use of this flag implies the provision of IPv6 prefix via SLAAC protocol, and the remaining parameters via DHCPv6 protocol.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] ipv6 nd other-config-flag
```

Parameters

None.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-LOOPBACK

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config)# ipv6 nd other-config-flag
```

ipv6 nd prefix

This command sets IPv6 prefix for broadcast SLAAC messages for the broadcast domain.

The use of a negative form (no) of the command.

Syntax

```
ipv6 nd prefix <IPV6-ADDR/LEN> <VLAID-TIME> <PREFER-TIME> [no-autoconfig] [no-onlink]
no ipv6 nd prefix
```

Parameters

<IPV6-ADDR/LEN> – IP address and prefix of a subnet, defined as X:X:X:X::X/EE where each X part takes values in hexadecimal format [0..FFFF] and EE takes values of [1..128];

<VLAID-TIME> – acceptable lifetime of the advertisement, specified in seconds [1..2147483647];

<PREFER-TIME> – preferable lifetime of the advertisement, specified in seconds [1..2147483647];

[no-autoconfig] – do not use Neighbor Discovery to set all valid channel prefixes from router advertisements (RA) received on the interface;

[no-onlink] – configures the specified prefix as non-on-link. The prefix will be declared with the L-bit set.

Default value

The prefix is not set;

valid time – 0;

preffered time – 0;

no-onlink – not set;

no-autoconfig – not set.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-LOOPBACK

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config)# ipv6 nd prefix
```

ipv6 nd ra hop-limit

This command sets hop-limit parameter for broadcast SLAAC messages for the broadcast domain.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 nd ra hop-limit <HOP-LIMIT>
no ipv6 nd ra hop-limit
```

Parameters

<HOP-LIMIT> – specifies in the range [0..255].

Default value

64

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-LOOPBACK

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config)# ipv6 nd ra hop-limit 128
```

ipv6 nd ra lifetime

This command sets lifetime parameter for broadcast SLAAC messages for the broadcast domain.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 nd ra lifetime <TIME>
no ipv6 nd ra lifetime
```

Parameters

<TIME> – lifetime of the advertised router advertisements is determined in seconds [0..9000].

Default value

Not limited.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-LOOPBACK

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config)# ipv6 nd ra lifetime 120
```

ipv6 nd ra max-interval

This command sets maximum interval for broadcast SLAAC messages for the broadcast domain.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 nd ra max-interval <TIME>
no ipv6 nd ra max-interval
```

Parameters

<TIME> – maximum interval for sending SLAAC messages is determined in seconds [4..1800].

Default value

600

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-LOOPBACK

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config)# ipv6 nd ra max-interval 1200
```

ipv6 nd ra min-interval

This command sets minimum interval for broadcast SLAAC messages for the broadcast domain.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 nd ra min-interval <TIME>
no ipv6 nd ra min-interval
```

Parameters

<TIME> – minimum interval for sending SLAAC messages is determined in seconds [3..1350].

Default value

200

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-LOOPBACK

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config)# ipv6 nd ra min-interval 30
```

ipv6 nd router-preference

This command sets the priority of the router in the process of selecting the default route for the broadcast domain.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 nd router-preference <ACTION>
no ipv6 nd router-preference
```

Parameters

<ACTION> – takes values:

- low;
- medium;

Default value

medium

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-LOOPBACK

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config)# ipv6 nd router-preference low
```

ipv6 redirects

The command enables the mechanism of sending ICMP messages about the existence of a router with a higher priority in the IPv6 network for a specific IPv6 destination.

The use of a negative form of the command (no) disables the mechanism of sending ICMP messages about the existence of a router with a higher priority in the IPv6 network for a specific IPv6 destination.

Syntax

`[no] ipv6 redirects`

Parameters

The command does not contain parameters.

Required privilege level

15

Default value

Sending of ICMP messages about the existence of a router with a higher priority in the IP network for a specific IP destination is enabled.

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

Example

```
esr(config-if-gi)# no ipv6 redirects
```

ipv6 route source-route

This command enables the ipv6 source-route option on the router.

The use of a negative form of the command (no) disables the support for the ipv6 source-route option.

Syntax

```
[no] ipv6 route source-route
```

Parameters

None.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 route source-route
```

ipv6 unreachables

The command enables sending of IPv6 CIMP packets about destination address unavailability.

The use of a negative form of the command (no) disables the sending of IPv6 ICMP packets about destination address unavailability.

Syntax

```
[no] ipv6 unreachables
```

Parameters

The command does not contain parameters.

Default value

Sending of IPv6 ICMP packets about destination address unavailability is enabled.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

Example

```
esr(config-if-gi)# no ipv6 unreachables
```

show ipv6 interfaces

The command displays information on IPv6 interfaces existing in the system.

Syntax

```
show ipv6 interfaces { { <IF> | <TUN> | ipv6-address <IPV6-ADDR> } | vrf <VRF> [ ipv6-
address <IPV6-ADDR> ] }
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, IPv6 interfaces will be displayed in a specified VRF;

<IF> – system interface names, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – names of tunnels are specified as described in section Types and naming order of router tunnels;

<IPV6-ADDR> – IPv6 address, defined as X:X:X:X::X/EE where each X part takes values in hexadecimal format [0..FFFF] and EE takes values of [1..128]. When specifying this parameter, IPv6 interface with a specified IPv6 address will be displayed.

You may specify several system interfaces in the command. If you do not specify interface indexes, then all IPv6 interfaces related to the system interfaces of the specified type will be displayed.

If the command specifies a certain system interface that receives IPv6 parameters via DHCP, DHCP client settings and the status of IPv6 parameters current lease will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 interfaces
IPv6 address                                          Interface       Type
---------------------------------------------------   -------------   -------
fc00::2/120                                           gi1/0/5         static
fe80::aaf9:4bff:feaa:45/64                            gi1/0/5         static
esr# show ip interfaces gigabitethernet 1/0/16
IP address            Interface       Type
-------------------   -------------   -------
16.0.0.2/24           gi1/0/16        DHCP
 DHCP Client settings:
 DHCP Server:          N/A
 Lease time(dd:hh:mm): 00:02:00
 Reboot time:          10 seconds
 Retry time:           300 seconds
 Timeout:              60 seconds
 Select timeout:       0 seconds
 Vendor class ID:      N/A
 Ignore options:
    router
 Latest lease contents:
 Lease time(dd:hh:mm): 00:02:00
 DHCP message type:    DHCPACK
 Renew at:             Wednesday2015/02/25 12:22:24 2015/02/25 12:22:24
 Rebind at:            Wednesday2015/02/25 13:14:09 2015/02/25 13:14:09
 Expires at:           Wednesday2015/02/25 13:29:09 2015/02/25 13:29:09
```

19 Profiles management

- address-port pair
- application
- category
- description
- email
- ip address-range
- ip prefix
- ipv6 address-range
- ipv6 prefix
- mac address
- object-group address-port
- object-group application
- object-group content-filter
- object-group email
- object-group mac
- object-group network
- object-group service
- object-group url
- port-range
- regexp
- show object-group
- url
- vendor

address-port pair

The command is used to set the bundle of IP addresses and TCP/UDP port.

The use of a negative form (no) of the command removes an entry from a configured profile.

Syntax

[no] address-port pair < ADRR >:< PORT >

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<PORT> – port number, takes values of [1..65535].

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-ADDRESS-PORT

Example

```
esr(config-object-group-address-port)# address-port pair 192.168.1.1:23
```

application

The command specifies the applications that are covered by the profile.

The use of a negative form (no) of the command removes an application from the current profile.

Syntax

```
[no] application < APPLICATION >
```

Parameters

<APPLICATION> – specifies the application covered by this profile.

May take the following values:

- afp – Apple Filing Protocol;
- amazon – Amazon Data Services;
- amqp – Advanced Message Queuing Protocol;
- apple – Apple Inc.;
- apple-icloud – Apple iCloud;
- apple-itunes – Apple iTunes;
- applejuice – Applejuice P2P;
- avi – AVI content in HTTP payload;
- ayiya – Anything In Anything;
- battlefield – Battlefield;
- bgp – Border Gateway Protocol;
- bittorrent – BitTorrent;
- bjnp – Canon BJNP protocol;
- cisco-skinny – Cisco Skinny;
- cisco-vpn – Cisco VPN;
- citrix – Citrix;
- citrix-online – Citrix-online;
- cloudflare – Cloudflare Inc.;
- coap – Constrained Application Protocol;
- collectd – Collectd;
- corba – Common Object Request Broker Architecture;
- dce-rpc – Distributed Computing Environment / Remote Procedure Calls;
- deezer – Deezer (music streaming service);
- dhcp – Dynamic Host Configuration Protocol;
- dhcpv6 – IPv6 Dynamic Host Configuration Protocol;
- directconnect – Direct Connect;
- dns – Domain Name System;
- dnscrypt – DNSCrypt;
- drda – Distributed Relational Database Architecture;
- dropbox – Dropbox;
- ebay – eBay;
- edonkey – eDonkey;
- egp – Exterior Gateway Protocol;

- epp – Extensible Provisioning Protocol.

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-APPLICATION

Example

```
esr(config-object-group-application)# application egp
```

category

This command is used to set the content filtering category.

The use of a negative form (no) of the command removes a specified category.

Syntax

```
[no] category <CATEGORY>
```

Parameters

<CATEGORY> – category name. May take the following values:

- abortions – abortion related content.
- addictions – alcohol, tobacco and drugs related content.
- ads – teaser network websites, advertising services, contextual advertising services, advertising agencies.
- adult-content – adult content. Pornographic and erotic content, nudity.
- adult-dating – adult dating websites.
- adware – content related to software that downloads or displays unwanted advertising, collects marketing data.
- alcohol – sale of alcohol, discussion, articles on the subject.
- animals – content related to pets and animals. Articles and discussions.
- anonymizers – web anonymizers and proxy servers.
- anorexia – content related to anorexia and its promotion.
- banks – bank websites.
- beauty – content related to beauty, health and wellness.
- blogs – blogs and blog hostings.
- books – online libraries, bookstores, book reviews.
- business – web pages of commercial firms, business associations, industry groups and corporate websites in general.
- casino – content related to casino and card games.
- chats – web chats, forums, instant messaging.
- commerce – content related to online transactions and commerce.
- communication – content and services related to communication.
- cultural-heritage – web pages containing information about and promoting the visual arts, performing arts, and cultural heritage.
- culture – content related to culture and society.

- cryptocurrency – web pages about cryptocurrency and cryptomining software.
- dating – dating services. Does not include adult dating services.
- discrimination – content related to hatred and discrimination.
- downloads – software, audio or video content downloads.
- dyn-dns – resource provides dynamic DNS service.
- educational-institutions – school and university websites.
- education – education related content.
- electronics – web pages with content related to consumer electronics.
- email – email web services.
- encyclopedias – educational portals, knowledge bases, encyclopedias.
- extremist-materials – content from the Federal List of Extremist Materials of the Russian Federation.
- family – content related to home and family. Related stores.
- fashion – fashion and style related content.
- file-sharing – file sharing sites and services.
- finance – web pages containing information about finance and the national economy.
- food – food related content.
- gambling – content and services related to gambling, lotteries, sweepstakes.
- games – content related to games.
- government – government websites, political parties, laws, and law firms.
- health – content related to health, medicine, healthy lifestyles, diets, vegetarians.
- hobbies – content related to hobbies and entertainment.
- hosting – hosting and domain registration services, whois services, domain sellers.
- humour – humorous content.
- hunting – content related to hunting and fishing.
- illegal-content – content that is prohibited by Russian law.
- internet-services – sites of global and local Internet service providers.
- it – content related to information technology.
- it-security – web pages containing information about IT security, including websites that provide security products and services to corporate and home users.
- kids-internet – child content.
- lgbt – lesbian, gay, bisexual and transgender (LGBT) content.
- lingerie – content depicting underwear.
- magic – content related to esotericism, astrology, horoscopes, divination, chiromancy, ufology, magic.
- malware – malicious resources or malware download URLs.
- military – content related to weapons or military equipment.
- motor-vehicles – web pages with content about automobile vehicles.
- music – music related content.
- narcotics – narcotics. Sales, production, discussion.
- news – news and media websites.
- nudism – nudism and exhibitionism, related communities.
- online-betting – online betting services. Betting on horse racing and sports.
- online-lotteries – online lotteries.
- online-payments – online payment and online banking services.
- pharmacy – content related to pharmacy, legal drugs, and medical supplements.
- phishing – phishing websites.
- porno – pornographic and erotic content, nudity.
- profane-language – content with profanity.
- racism – extremist and racist content.
- rat-node – resources with software for remote administration.
- recruitment – recruitment, hiring and employment websites.
- religion – content related to religious associations, organizations, cults and sects.
- rental-services – real estate rental services.
- riskware – content related to legitimate programs that can cause damage if used by malicious users.
- roskomnadzor – content blacklisted by Roskomnadzor.
- searchers – search engines and services.

- sex-education – content related to sex education.
- sex-shops – sex shops and sex toys.
- shops – online stores and auctions.
- social-networks – social networks.
- spam – sites specifically created for spam purposes and/or found in spam emails.
- sport – content related to sports and bodybuilding.
- streams – video and audio hosting, downloads, streaming.
- self-damage – content has to do with suicide, self-harm, and self-mutilation.
- tobacco – tobacco sales, discussion.
- torrent – torrent trackers and forums, including protocol specifications and software.
- tor – Tor network nodes.
- traveling – travel related content.
- tv-radio – content related to television and radio.
- uncategorized – content that does not fall into any category.
- violence – physical or psychological violence, cruelty to animals.
- weapons – content relating to weapons, explosives and pyrotechnic products.
- 436-fz – content prohibited by Federal Law of the Russian Federation No. 436-FZ.

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-CF-KASPERSKY

Example

```
esr(config-object-group-cf-kaspersky)# category books
```

description

The command is used to change a profile description.

The use of a negative form (no) of the command removes a profile description.

Syntax

`description <DESCRIPTION>`

`no description`

Parameters

<DESCRIPTION> – profile description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-NETWORK

CONFIG-OBJECT-GROUP-SERVICE

CONFIG-OBJECT-GROUP-MAC

CONFIG-OBJECT-GROUP-APPLICATION

CONFIG-OBJECT-GROUP-URL

CONFIG-OBJECT-GROUP-ADDRESS-PORT

CONFIG-OBJECT-GROUP-CONTENT-FILTER

CONFIG-OBJECT-GROUP-MAIL

Example

Set the description for IP addresses profile:

```
esr(config-object-group-network)# description "Internal addresses"
```

email

The command is used to specify a mail domain or mailbox address.

The use of a negative form (no) of the command removes an entry from a configured profile.

Syntax

`[no] email <NAME>`

Parameters

<NAME> – mail domain or mailbox addresses, specified by a string of 1 to 63 characters.

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-MAIL

Example

```
esr(config-object-group-email)# email eltex@eltex-co.ru
esr(config-object-group-email)# email eltex-co.ru
```

ip address-range

The command specifies IP addresses range.

The use of a negative form (no) of the command removes an entry from a configured profile.

Syntax

```
[no] ip address-range <FROM-ADDR>[-<TO-ADDR>]
```

Parameters

<FROM-ADDR> – range starting IP address;

<TO-ADDR> – range ending IP address, optional parameter; If the parameter is not specified, a single IP address is set by the command.

The addresses are defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

You can specify up to 64 different IP ranges within one address group for ESR-20/21/100/200/1000/1200/1500/1700

You can specify up to 6 different IP ranges within one address group for ESR-10/12V/12VF/14VF

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-NETWORK

Example

```
esr(config-object-group-network)# ip address-range 192.168.1.1 192.168.1.25
```

ip prefix

The command specifies a subnet.

The use of a negative form (no) of the command removes a specified subnet.

Syntax

```
[no] ip prefix <ADDR/LEN>
```

Parameters

<ADDR/LEN> – IP subnet, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32].

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-NETWORK

Example

```
esr(config-object-group-network)# ip prefix 10.10.10.0/24
```

ipv6 address-range

The command specifies IPv6 addresses range. The use of a negative form (no) of the command removes an entry from a configured profile.

Syntax

```
[no] ipv6 address-range <FROM-ADDR>[-<TO-ADDR>]
```

Parameters

<FROM-ADDR> – range starting IPv6 address.

<TO-ADDR> – range ending IPv6 address, optional parameter. If the parameter is not specified, a single IPv6 address is set by the command.

The addresses are defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

You can specify up to 64 different IP ranges within one address group for ESR-20/21/100/200/1000/1200/1500/1700

You can specify up to 6 different IP ranges within one address group for ESR-10/12V/12VF/14VF

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-NETWORK

Example

```
esr(config-object-group-network)# ipv6 address-range fc00::1:1-fc00:1::32
```

ipv6 prefix

The command specifies IPv6 subnet.

The use of a negative form (no) of the command removes a specified subnet.

Syntax

```
[no] ipv6 prefix <IPV6-ADDR/LEN>
```

Parameters

<IPV6-ADDR/LEN> – IP address and mask of a subnet, defined as X:X:X:X::X/EE where each X part takes values in hexadecimal format [0..FFFF] and EE takes values of [1..128].

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-NETWORK

Example

```
esr(config-object-group-network)# ipv6 prefix fc00::/126
```

mac address

The command specifies MAC addresses range.

The use of a negative form (no) of the command removes an entry from a configured profile.

Syntax

```
[no] mac address <MAC-ADDR> <MAC-MASK>
```

Parameters

<MAC-ADDR> – MAC address, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF];

<MAC-MASK> – MAC address mask, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF]. Mask bits, set to zero, specify MAC address bits excluded from the comparison when searching. Mask default value: FF:FF:FF:FF:FF:FF.

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-MAC

Example

```
esr(config-object-group-mac)# mac address a8:f9:4b:80:e7:00 FF:FF:FF:FF:FF:00
```

object-group address-port

This command creates a profile of bundles of IP addresses and TCP/UDP ports. Profiles are used to configure services that work with pools of IP addresses and TCP/UDP ports – NAT, Firewall.

The use of a negative form (no) of the command removes IP address profile.

Syntax

```
[no] object-group address-port <NAME>
```

Parameters

<NAME> – the name of the configured profile of IP address bundles and TCP/UDP ports is specified by a string of up to 31 characters. The use of a negative form (no) of the command with «all» parameter removes all IP address profiles.

Required privilege level

10

Command mode

CONFIG

Example

To create IP addresses profile with name *remote* and to switch to profile configuration mode:

```
esr(config)# object-group address-port WEB
```

object-group application

The command is used to create application profile. The profile is used for filtration on the basis of applications (DPI).

The use of a negative form (no) of the command removes the profile.

Syntax

```
[no] object-group application <NAME>
```

Parameters

<NAME> – application profile name, set by the string of up to 31 characters. When removing, you can use ‘all’ key instead of the name. When using the «all» key, all application profiles will be removed.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# object-group application OGA045
```

object-group content-filter

This command is used to create a content filtering category profile. This profile is used in a set of custom rules.

The use of a negative form (no) of the command removes the profile.

Syntax

```
[no] object-group content-filter <NAME>
```

Parameters

<NAME> – name of the content filtering profile, specified as a string of up to 31 characters. When removing, you can use ‘all’ key instead of the name. When using the «all» key, all content filtering profiles will be removed.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# object-group content-filter OGC042
```

object-group email

This command is used to create a profile of mail domains and mailbox addresses. This profile is used in the "Antispam" service profile rules.

The use of a negative form (no) of the command removes the profile.

Syntax

```
[no] object-group email <NAME>
```

Parameters

<NAME> – profile name of mail domains and mailbox addresses, specified by a string of up to 31 characters. When removing, you can use ‘all’ key instead of the name. Using the "all" key will delete all mail domain profiles and mailbox addresses.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# object-group email chinese_spammers
```

object-group mac

The command is used to create MAC address profile. The profile is used in MAC-based VLAN.

The use of a negative form (no) of the command removes the profile.

Syntax

```
[no] object-group mac <NAME>
```

Parameters

<NAME> – MAC addresses profile name, set by the string of up to 31 characters. When removing, you can use ‘all’ key instead of the name. When using the «all» key, all MAC address profiles will be removed.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# object-group mac OGM007
```

object-group network

The command is used to create IP address profile. The profiles are used to configure services operating with IP address pools – for example, NAT, Firewall, Remote-Access – as well as to create prefix list.

The use of a negative form (no) of the command removes IP address profile.

Syntax

```
[no] object-group network <NAME>
```

Parameters

<NAME> – configured IP addresses profile, set by the string of up to 31 characters. The use of a negative form (no) of the command with ‘all’ parameter removes all IP address profiles.

Required privilege level

10

Command mode

CONFIG

Example

To create IP addresses profile with name *remote* and to switch to profile configuration mode:

```
esr(config)# object-group network remote
```

object-group service

The command creates TCP/UDP ports profile. The profile is used in NAT and Firewall services rules.

The use of a negative form (no) of the command removes the profile.

Syntax

```
[no] object-group service <NAME>
```

Parameters

<NAME> – port profile name, set by the string of up to 31 characters. The use of a negative form (no) of the command with ‘all’ parameter removes all TCP/UDP ports profiles.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# object-group service ssh
```

object-group url

The command is used to create URL link profile.

The use of a negative form (no) of the command removes the profile.

Syntax

```
[no] object-group url <NAME>
```

Parameters

<NAME> – port profile name, set by the string of up to 31 characters. The use of a negative form (no) of the command with 'all' parameter removes all URL links profiles.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# object-group url vk
```

port-range

The command specifies the range of TCP/UDP ports related to the profile.

The use of a negative form (no) of the command removes an entry from a configured profile.

Syntax

```
port-range <PORT>
no port-range [<PORT> | all]
```

Parameters

<PORT> – port number, takes values of [1..65535].

You can specify several ports separated by commas ',' or you can specify the range of ports with '-'. Example of the record: <PORT>, <PORT> or <PORT>-<PORT> or <PORT>-<PORT>, <PORT>-<PORT>.

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-SERVICE

Example

```
esr(config-object-group-service)# port-range 22
```

regexp

This command describes a URL link pattern.

The use of a negative form (no) of the command removes a URL link pattern.

Syntax

```
regexp <REGEXP>
no regexp {<REGEXP>|all}
```

Parameters

<REGEXP> – regular expression. Described by the string of up to 255 characters. The character '\' must be shielded.

all — the key used to delete all created rules.

Default value

Pattern is not created.

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-URL

Example

```
esr(config-object-group-url)# '^http:\/\/site\.ru'
```

show object-group

The command displays information on IP addresses and TCP/UDP ports profiles.

Syntax

```
show object-group <PROFILE_TYPE> [<NAME>]
```

Parameters

<PROFILE_TYPE> – profile type:

- network – IP addresses profile;
- service – TCP/UDP ports profile;

<NAME> – profile name, set by the string of up to 31 characters, optional parameter. If profile name is not specified, information on all IP addresses and TCP/UDP ports profiles will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show object-group network
Network                            Description
--------------------------------   --------------------------------
remote                             --
local                              --
tunnel                             --
esr# show object-group network remote
IP Addresses
--------------------------------
10.102.0.0/16
esr# show object-group service
Service                            Description
--------------------------------   --------------------------------
telnet                             --
ssh                                --
dhcp_server                        --
dhcp_client                        --
ntp                                --
esr# show object-group service ssh
Port ranges
--------------------------------
22
```

url

The command specifies URL link.

The use of a negative form (no) of the command removes a link from a configured profile.

Syntax

```
url <URL>
no url [ <URL> | all ]
```

Parameters

<URL> – text field containing URL link of 8-255 characters length.

When removing it with ‘all’ key, all previously added URL links will be removed.

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-URL

Example

```
esr(config-object-group-url)# url https://vk.com
```

vendor

This command is used to set the content filtering category provider.

The use of a negative form (no) of the command removes the content filtering categories of this vendor.

Syntax

```
[no] vendor <CONTENT-FILTER-VENDOR>
```

Parameters

<CONTENT-FILTER-VENDOR> – name of the content filtering category provider. Takes the following values:

Kaspersky-Lab – in the current version of the software, only Kaspersky Lab can act as a content filtering category provider.

Required privilege level

10

Command mode

CONFIG-OBJECT-GROUP-CONTENT-FILTER

Example

```
esr(config-object-group-content-filter)# vendor kaspersky-lab
```

20 NAT management

- action destination-nat
- action source-nat
- description
- enable
- from
- ip address
- ip address-range
- ip nat proxy-arp
- ip port
- ip port-range
- match destination-address
- match destination-address-port
- match destination-port
- match icmp
- match protocol
- match source-address
- match source-address-port
- match source-port
- nat alg
- nat destination
- nat source
- persistent
- pool
- rearrange
- renumber rule
- rule
- ruleset
- show ip nat alg
- show ip nat pool
- show ip nat ruleset
- show ip nat translations
- show ip nat proxy-arp
- to

action destination-nat

This command performs translation of the address and port of the recipient for traffic that meets the specified criteria.

The use of a negative form (no) of the command sets the default value.

Syntax

```
action destination-nat { off | pool <NAME> | netmap <ADDR/LEN> }
```

```
no action destination-nat
```

Parameters

off – translation is disabled. Traffic that falls under the specified criteria will not be changed;

pool<NAME> – name of the pool that contains IP addresses and/or TCP/UDP ports set; For traffic that falls under the specified criteria, the IP address and TCP/UDP port of the recipient will be changed to values selected from the pool;

netmap <ADDR/LEN> – IP subnet used for broadcast. For traffic that falls under the specified criteria, the recipient's IP address will be changed to an IP address from the specified subnet. The parameter is defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32].

Default value

None

Required privilege level

10

Command mode

CONFIG-DNAT-RULE

Example

```
esr(config-dnat-rule)# action destination-nat netmap 10.10.10.0/24
```

action source-nat

This command specifies the 'translation of source address and port' action type for the traffic meeting the requirements of 'match' command.

The use of a negative form (no) of the command sets the default value.

Syntax

```
action source-nat { off | pool <NAME> | netmap <ADDR/LEN> [static] | interface
[FIRST_PORT - LAST_PORT] }
```

```
no action source-nat
```

Parameters

off – translation is disabled. Traffic that falls under the specified criteria will not be changed;

pool<NAME> – names the pool that contains IP addresses and/or TCP/UDP ports set; For traffic that falls under the specified criteria, the IP address and TCP/UDP port of the sender will be changed to values selected from the pool;

netmap <ADDR/LEN> – sets IP subnet used for broadcast. For traffic that falls under the specified criteria, the sender's IP address will be changed to an IP address from the specified subnet. The parameter is defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32];

static - enable static NAT, available when using netmap;

interface [FIRST_PORT – LAST_PORT] – specify the translation to the interface IP address. For traffic that falls under the specified criteria, the IP address of the sender will be changed to the IP address of the interface to which this traffic was transmitted. If the range of TCP/UDP ports is additionally specified, the translation will also occur for the source TCP/UDP ports, the will be replaced to the specified range.

Default value

None

Required privilege level

10

Command mode

CONFIG-SNAT-RULE

Example

```
esr(config-snat-rule)# action source-nat netmap 10.10.10.0/24
```

description

This command sets the description.

The use of a negative form (no) of the command removes description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – interface description, set by a string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

CONFIG-DNAT-POOL

CONFIG-SNAT-POOL

Example

```
esr(config-snat-ruleset)# description "test ruleset"
```

enable

The command enables a configurable rule.

The use of a negative form (no) of the command disables the usage of a configurable rule.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Example

```
esr(config-snat-rule)# enable
```

from

This command restricts field of use of the rule group. The rules will be applied only to traffic coming from a certain zone or interface.

The use of a negative form (no) of the command removes restriction of field of use of the rule group.

Syntax

```
from { zone <NAME> | interface <IF> | tunnel <TUN> | default }
no from
```

Parameters

<NAME> – isolation zone name;

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

default – denotes a group of rules for all traffic, the source of which did not fall under the criteria of other groups of rules.

> ⚠ The rule group with the 'default' value of the 'from' parameter can be only one.

Default value

None

Required privilege level

10

Command mode

CONFIG-DNAT-RULESET

Example

```
esr(config-dnat-ruleset)# from zone untrusted
```

ip address

This command sets the internal IP address which will replace a destination IP address.

The use of a negative form (no) of the command removes a specified IP address.

Syntax

```
ip address <ADDR>
no ip address
```

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-DNAT-POOL

Example

```
esr(config-dnat-pool)# ip address 10.10.10.10
```

ip address-range

Set the range of external IP addresses which will replace a source IP address.

The use of a negative form (no) of the command removes a specified range of addresses.

Syntax

```
ip address-range <IP>[-<ENDIP>]
```

```
no ip address-range
```

Parameters

<IP> – IP address of the beginning of the range, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<ENDIP> – IP address of the end of the range, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. If IP address of the end of the range is not specified, only IP address of the beginning of the range is used as IP address for translation.

Required privilege level

10

Command mode

CONFIG-SNAT-POOL

Example

```
esr(config-snat-pool)# ip address-range 10.10.10.1-10.10.10.20
```

ip nat proxy-arp

This command allows the router to respond to the ARP requests for IP addresses from a specified pool. The function is necessary in order not to assign all IP addresses from the translation pool on the interface.

Syntax

```
ip nat proxy-arp <OBJ-GROUP-NETWORK-NAME>
```

```
no ip nat proxy-arp
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – IP addresses profile name, set by the string of up to 31 characters.

Default value

NAT Proxy ARP function is disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-CELLULAR-MODEM

CONFIG-LT

Example

```
esr(config-if-gi)# ip nat proxy-arp nat-pool
```

ip port

Sets the internal TCP/UDP port which will replace a destination TCP/UDP port.

The use of a negative form (no) of the command removes a specified TCP/UDP port.

Syntax

```
ip port <PORT>
no ip port
```

Parameters

<PORT> – TCP/UDP port, takes values of [1..65535].

Required privilege level

10

Command mode

CONFIG-DNAT-POOL

Example

```
esr(config-dnat-pool)# ip port 5000
```

ip port-range

Specify the range of external TCP/UDP ports which will replace a source TCP/UDP port.

The use of a negative form (no) of the command removes a specified range of ports.

Syntax

```
ip port-range <PORT>[-<ENDPORT>]
```

```
no ip port-range
```

Parameters

<PORT> – TCP/UDP port of the beginning of range, takes values of [1..65535];

<ENDPORT> – TCP/UDP port of the end of range, takes values of [1..65535]. If TCP/UDP port of the end of the range is not specified, only TCP/UDP port of the beginning of the range is used as TCP/UDP port for translation.

Required privilege level

10

Command mode

CONFIG-SNAT-POOL

Example

```
esr(config-snat-pool)# ip port-range 20-100
```

match destination-address

Set the profile of destination IP addresses for which the rule should work.

When using 'not' command, the rule will work for destination IP addresses which are not included in a specified profile. The use of a negative form (no) of the command cancels set action.

Syntax

```
match [not] destination-address <OBJ-GROUP-NETWORK-NAME>
```

```
no match destination-address
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – IP addresses profile name, set by the string of up to 31 characters. When specifying the 'any' value, the rule will be triggered for any source IP address.

Default value

any

Required privilege level

10

Command mode

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Example

```
esr(config-snat-rule)# match destination-address remote
```

match destination-address-port

This command sets the profile of IP address bundles and destination TCP/UDP ports for which the rule should work.

When using 'not' parameter, the rule will work for IP address bundles and destination TCP/UDP ports which are not included in a specified profile.

The use of a negative form (no) of the command cancels the assignment.

Syntax

`match [not] destination-address-port <OBJ-GROUP-ADDRESS-PORT-NAME>`

`no match destination-address`

Parameters

<OBJ-GROUP-ADDRESS-PORT-NAME> – the name of the profile of IP address bundles and TCP/UDP ports is specified by a string of up to 31 characters. When specifying the value 'any', the rule will not consider this filtering method.

Default value

any

Required privilege level

10

Command mode

CONFIG-SNAT-RULE

Example

```
esr(config-snat-rule)# match destination-address local
```

match destination-port

This command sets the profile of destination TCP/UDP ports for which the rule should work.

When using 'not' command, the rule will work for destination TCP/UDP ports which are not included in a specified profile.

The use of a negative form (no) of the command cancels set action.

Syntax

```
match [not] destination-port <PORT-SET-NAME>
```

```
no match destination-port
```

Parameters

<PORT-SET-NAME> – port profile name, set by the string of up to 31 characters. When specifying the 'any' value, the rule will be triggered for any source TCP/UDP port.

Required privilege level

10

Command mode

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Example

```
esr(config-snat-rule)# match destination-port ssh
```

match icmp

The command is used to configure ICMP parameters if it is selected by 'match protocol' command. It specifies the type and code of ICMP messages for which the rule should work.

When using 'not' command, the rule will work for all types and codes of ICMP messages excluding specified ones.

The use of a negative form (no) of the command sets the default value.

Syntax

```
match [not] icmp { <ICMP_TYPE> <ICMP_CODE> | <OPTION> }
```

```
no match icmp
```

Parameters

<ICMP_TYPE> – ICMP message type, takes values of [0..255];

<ICMP_CODE> – ICMP message code, takes values of [0..255]. When specifying the “any” value, the rule will work for any ICMP message code;

<OPTION> – standard types of ICMP messages can take values:

- administratively-prohibited;
- alternate-address;
- conversion-error;
- dod-host-prohibited;
- dod-network-prohibited;
- echo;
- echo-reply;
- host-isolated;
- host-precedence;
- host-redirect;
- host-tos-redirect;
- host-tos-unreachable;
- host-unknown;
- host-unreachable;
- information-reply;
- information-request;
- mask-reply;
- mask-request;
- network-redirect;
- network-tos-redirect;
- network-tos-unreachable;
- network-unknown;
- network-unreachable;
- option-missing;
- packet-too-big;
- parameter-problem;
- port-unreachable;
- precedence;
- protocol-unreachable;
- reassembly-timeout;
- router-advertisement;
- router-solicitation;
- source-quench;
- source-route-failed;
- time-exceeded;
- timestamp-reply;
- timestamp-request;
- traceroute.

Default value

any any

Required privilege level

10

Command mode

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Example

```
esr(config-snat-rule)# match icmp 2 any
```

match protocol

The command sets name or number of IP for which the rule should work.

When using 'not' parameter, the rule will work for all protocols except a specified one.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] protocol <TYPE>
no match protocol
match [not] protocol-id <ID>
no match protocol-id
```

Parameters

<TYPE> – protocol type, takes the following values: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre.

When specifying the 'any' value, the rule will work for any protocols.

<ID> – IP identification number, takes values of [0x00-0xFF].

Default value

any

Required privilege level

10

Command mode

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Example

```
esr(config-snat-rule)# match protocol udp
```

match source-address

The command specifies the profile of source IP addresses for which the rule should work.

When using 'not' command, the rule will work for destination IP addresses which are not included in a specified profile.

The use of a negative form (no) of the command cancels set action.

Syntax

```
match [not] source-address <OBJ-GROUP-NETWORK-NAME>

no match source-address
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – IP addresses profile name, set by the string of up to 31 characters. When specifying the 'any' value, the rule will be triggered for any source IP address.

Default value

any

Required privilege level

10

Command mode

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Example

```
esr(config-snat-rule)# match source-address local
```

match source-address-port

This command sets the profile of IP address bundles and source TCP/UDP ports for which the rule should trigger.

When using 'not' (match not) parameter, the rule will work for IP address bundles and source TCP/UDP ports which are not included in a specified profile.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] source-address-port <OBJ-GROUP-ADDRESS-PORT-NAME>
no match source-address-port <OBJ-GROUP-ADDRESS-PORT-NAME>
```

Parameters

<OBJ-GROUP-ADDRESS-PORT-NAME> – the name of the profile of IP address bundles and TCP/UDP ports is specified by a string of up to 31 characters. When specifying the value 'any', the rule will not consider this filtering method.

Default value

any

Required privilege level

10

Command mode

CONFIG-SNAT-RULE

Example

```
esr(config-snat-rule)# match source-address-port admin
```

match source-port

The command sets the profile of source TCP/UDP ports for which the rule should work.

When using 'not' command, the rule will work for source TCP/UDP ports which are not included in a specified profile. The use of a negative form (no) of the command cancels set action.

Syntax

```
match [not] source-port <PORT-SET-NAME>
no match source-port
```

Parameters

<PORT-SET-NAME> – port profile name, set by the string of up to 31 characters. When specifying the 'any' value, the rule will be triggered for any source TCP/UDP port.

Required privilege level

10

Command mode

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Example

```
esr(config-snat-rule)# match source-port telnet
```

nat alg

This command enables the IP address translation feature in the application level headers.

The use of a negative form (no) of the command disables the IP address translation feature in the application level headers.

Syntax

```
[no] nat alg { <PROTOCOL> }
```

Parameters

<PROTOCOL> is an application-level protocol, in which headers address translation should work, takes the values [ftp, h323, pptp, netbios-ns, gre, sip, tftp].

Instead of a certain protocol you can use the 'all' key that enables IP address translation in all available protocols headers.

Default value

The IP address translation feature in application level headers is disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# nat alg ftp
```

nat destination

This command allows you to enter the configuration mode of the destination address translation service (DNAT, Destination NAT).

The use of a negative form (no) of the command removes the configuration of the destination address translation service (DNAT, Destination NAT).

Syntax

```
[no] nat destination
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# nat destination
esr(config-dnat)#
```

nat source

This command allows you to enter the configuration mode of the source address translation service (SNAT, Source NAT).

The use of a negative form (no) of the command removes the configuration of the source address translation service (SNAT, Source NAT).

Syntax

```
[no] nat source
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# nat source
esr(config-snat)#
```

persistent

This command enables NAT persistent feature.

NAT persistent allows applications to use STUN (session traversal utilities for NAT) to establish a connection with devices behind the NAT gateway. This ensures that requests from the same internal address are translated to the same external address.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] persistent
```

Parameters

The command does not contain parameters.

Default value

NAT persistent feature disabled.

Required privilege level

10

Command mode

CONFIG-SNAT-POOL

Example

```
esr(config-snat-pool)# persistent
```

pool

The command creates and assigns a pool of IP addresses and TCP / UDP ports with a specific name for the NAT service and changes the command mode to SNAT POOL or DNAT POOL.

> ⚠ If a pool is used in any group of rules, then it cannot be deleted.

The use of a negative form (no) of the command removes a specified NAT addresses pool.

Syntax

```
[no] pool <NAME>
```

Parameters

<NAME> – NAT addresses pool name, set by the string of up to 31 characters. If you use the command to delete, then specifying the value 'all' will delete all pools of IP addresses and TCP/UDP ports.

Required privilege level

10

Command mode

CONFIG-DNAT

CONFIG-SNAT

Example

```
esr(config-snat)# pool nat
esr(config-snat-pool)#
```

rearrange

This command changes the step between the created rules.

Syntax

```
rearrange <VALUE>
```

Parameters

<VALUE> – maximum between rules, takes values of [1..50].

Required privilege level

10

Command mode

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

Example

```
esr(config-dnat-ruleset)# rearrange 10
```

renumber rule

This command changes the rule number.

Syntax

```
renumber rule <CUR_ORDER> <NEW_ORDER>
```

Parameters

<CUR_ORDER> – current rule number, takes values of [1..10000].

<NEW_ORDER> – new rule number, takes values of [1..10000].

Required privilege level

10

Command mode

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

Example

```
esr(config-dnat-ruleset)# renumber rule 13 100
```

rule

This command creates a rule with a specific number and sets the command interface mode SNAT RULE or DNAT RULE. The rules are proceeded by the device in number ascending order.

The use of a negative form of the command (no) removes the rule by number or all rules.

Syntax

`[no] rule <ORDER>`

Parameters

<ORDER> – rule number, takes values of [1..10000]. If the command is used for removal, when specifying the 'all' value all rules will be removed.

Required privilege level

10

Command mode

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

Example

```
esr(config-snat-ruleset)# rule 10
esr(config-snat-rule)#
```

ruleset

This command is used to create a group of rules with a specific name and to enter the SNAT RULESET or DNAT RULESET command mode.

The use of a negative form (no) of the command removes a specified rule group.

Syntax

`[no] ruleset <NAME>`

Parameters

<NAME> – rule group name, set by the string of up to 31 characters. If the command is used for removal, when specifying the 'all' value all rule groups will be removed.

Required privilege level

10

Command mode

CONFIG-DNAT

CONFIG-SNAT

Example

```
esr(config-snat)# ruleset wan
esr(config-snat-ruleset)#
```

show ip nat alg

This command displays information about the functionality of IP address translation in application level headers.

Syntax

```
show ip nat alg
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip nat alg
ALG Status:
    FTP:   Enabled
    H.323: Disabled
    GRE:   Disabled
    PPTP:  Disabled
    SIP:   Disabled
    SNMP:  Disabled
    TFTP:  Disabled
```

show ip nat pool

This command displays pools of internal and external IP addresses and TCP/UDP ports.

Syntax

```
show ip nat <TYPE> pools
```

Parameters

<TYPE> – the type of pools to view:

- source – external IP addresses and TCP/UDP ports;
- destination – inxternal IP addresses and TCP/UDP ports;

Required privilege level

1

Command mode

ROOT

Example

```
esr# show nat source pools
   Pools
   ~~~~~
ID     Name                    Ip address          Port      Description   Persi
                                                   range                   stent
----   ---------------------   -----------------   -------   -----------   -----
0      outside                 25.56.48.11         2000 -    outside-poo   false
                                                   3000       l
```

show ip nat ruleset

This command scans all or selected groups of rules used by the NAT function.

Syntax

```
show ip nat <TYPE> ruleset [<NAME>]
```

Parameters

<TYPE> – rule group type:

- source – rule group for sender's IP address and TCP/UDP port translation;
- destination – rule group for receiver's IP address and TCP/UDP port translation;

[NAME] – rule group name, optional parameter. If the name is not specified, a list of all rule groups will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip nat source rulesets
   Rulesets
   ~~~~~~~~
ID     Name                               To                  Description
----   --------------------------------   ------------------  -----------------
0      factory                            zone 'untrusted'
1      test                               gigabitethernet     test
                                            1/0/1
esr# show ip nat source rulesets factory
Ruleset:           factory
Description:
To:                none
Rules:
------
Order:             10
Description:       replace 'source ip' by outgoing interface ip address
Matching pattern:
    Protocol:      any(0)
    Src-addr:      any
    Dest-addr:     any
Action:          interface port any
Status:            Enabled
--------------------------------------------------------------------------------
```

show ip nat translations

This command shows broadcast sessions. To view information about statistics, you should enable counters (see section IP firewall mode).

Syntax

```
show ip nat translations [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source-address
<ADDR> ] [ outiside-source-address <ADDR> ] [ inside-destination-address <ADDR> ]
[ outside-destination-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-source-
port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port <PORT> ]
[ summary ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, NTP configuration will be displayed in a specified VRF.

summary – displays summary statistics for translation sessions;

<TYPE> – protocol type, takes the following values: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<PORT> – TCP/UDP port, takes values of [1..65535];

For Source NAT:

- inside-source-address – key to specify the source IP address before translation;
- inside-destination-address – key to specify destination IP address on routers input;
- outside-source-address – key to specify the source IP address after translation;
- outside-destination-address – key to specify destination IP address on routers output;
- inside-source-port – key to specify source TCP/UDP port before translation;
- outside-source-port – key to specify source TCP/UDP port after translation;
- inside-destination-port – key to specify destination TCP/UDP port before translation;
- outside-destination-port – key to specify destination TCP/UDP port after translation;

For Destination NAT

- inside-source-address – key to specify source IP address on routers output;
- inside-destination-address – key to specify destination IP address after translation;
- outside-source-address – key to specify source IP address on routers input;
- outside-destination-address – key to specify destination IP address before translation;
- inside-source-port – key to specify source TCP/UDP port before translation;
- outside-source-port – key to specify source TCP/UDP port after translation;
- inside-destination-port – key to specify destination TCP/UDP port before translation;
- outside-destination-port – key to specify destination TCP/UDP port after translation;

Required privilege level

1

Command mode

ROOT

Example 1

Source NAT

```
esr# show ip nat translations
Prot   Inside source  Inside destination Outside source Outside destination Pkts Bytes
----   ------------  -----------------   ------------   --------------     -----  -----
icmp   115.0.0.10         1.1.0.2         1.1.0.24          1.1.0.2          3    252
```

Example 2

Destination NAT

```
esr#  show ip nat translations
Prot   Inside source  Inside destination Outside source Outside destination Pkts Bytes
----   ------------  -----------------   ------------   --------------     -----  -----  icmp
1.1.0.2        115.0.0.10         1.1.0.2         1.1.0.16          --    --
```

show ip nat proxy-arp

This command displays the NAT Proxy ARP settings.

Syntax

```
show ip nat proxy-arp
```

Required privilege level

1

Command mode

ROOT

Example

```
esr# show nat proxy-arp
Interface     IP address range
-----------   ---------------------------------------------
gi1/0/15      115.0.0.15-115.0.0.100
```

to

This command restricts field of use of the rule group. The rules will be applied only to traffic coming to a certain zone or interface.

The use of a negative form (no) of the command removes restriction of field of use of the rule group.

Syntax

```
to { zone <NAME> | interface <IF> | tunnel <TUN> | default }
no to
```

Parameters

<NAME> – isolation zone name;

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

default - denotes a group of rules for all traffic, the destination point of which did not fall under the criteria of other groups of rules.

> ⚠ The rule group with the 'default' value of the 'to' parameter can be only one.

Default value

None

Required privilege level

10

Command mode

CONFIG-SNAT-RULESET

Example

```
esr(config-snat)# ruleset test
esr(config-snat-ruleset)# to interface gigabitethernet 1/0/1
```

21 IPsec VPN configuration

- VPN management IKE configuration
    - access profile
    - address-assignment pool
    - assign-interface
    - authentication algorithm
    - authentication mode
    - authentication method
    - bind-interface vti
    - certificate
    - data-tunnel address
    - dead-peer-detection action
    - dead-peer-detection interval
    - dead-peer-detection timeout
    - description
    - dh-group
    - encryption algorithm
    - ike-policy
    - ip prefix
    - lifetime seconds
    - local address
    - local interface
    - local network
    - management-tunnel address
    - mobike disable
    - mode
    - mode
    - password
    - password local-crt-key
    - pfs dh-group
    - pre-shared-key
    - proposal
    - remote address
    - remote network
    - remote network dynamic client
    - security ike gateway
    - security ike policy
    - security ike proposal
    - security ike session uniqueids
    - show security ike
    - user
    - version
    - xauth access-profile
- VPN management. IPsec configuration
    - authentication algorithm
    - description
    - enable
    - encryption algorithm
    - ike dscp
    - ike establish-tunnel
    - ike gateway
    - ike idle-time
    - ike rekey disable

- ike rekey margin
- ike rekey randomization
- ike ipsec-policy
- lifetime
- manual authentication algorithm
- manual authentication key
- manual bind-interface vti
- manual encryption algorithm
- manual encryption key
- manual mode
- manual protocol
- manual spi
- mode
- proposal
- protocol
- security ipsec policy
- security ipsec proposal
- security ipsec vpn
- show security ipsec
- show security ipsec vpn authentication
- show security ipsec vpn status

VPN management IKE configuration

access profile

The command creates user's configuration profile for IKE-GATEWAY with a certain name and switches to profile configuration mode.

The use of a negative form (no) of the command removes a specified user's configuration profile for IKE-GATEWAY.

Syntax

```
[no] access profile <NAME>
```

Parameters

<NAME> – IKE-GATEWAY user profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# access profile OFFICE
```

address-assignment pool

The command creates a pool of addresses and configure parameters for the dynamic configuration of IPsec clients.

The use of a negative form (no) of the command removes address pool.

Syntax

```
[no] address-assignment pool <NAME>
```

Parameters

<NAME> – addresses pool name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# address-assignment pool CENTER
esr(config-pool)#
```

assign-interface

This command specifies a loopback interface for assigning a dynamic address received from an IPsec-VPN server.

The use of a negative form of the command (no) removes the loopback interface to assign a dynamic address received from the IPsec-VPN server.

Syntax

```
assign-interface loopback <LOOPBACK>[-<LOOPBACK>
```

```
no assign-interface
```

Parameters

<LOOPBACK> – number of the loopback interface created earlier, takes a value in the range [1..8].

Default value

None

Required privilege level

10

Command mode

CONFIG-IKE-GW

Example

```
esr(config-ike-gw)# assign-interface loopback 3
```

authentication algorithm

This command sets the authentication algorithm that is used to authenticate messages of an established IKE connection. When establishing an IKE connection, message authentication by key is used (authentication, see section password).

The use of a negative form (no) of the command sets the default value.

Syntax

```
authentication algorithm <ALGORITHM>
no authentication algorithm
```

Parameters

<ALGORITHM> – authentication algorithm, takes values of: md5, sha1, sha2-256, sha2_384, sha2-512.

Default value

sha1

Required privilege level

15

Command mode

CONFIG-IKE-PROPOSAL

Example

```
esr(config-ike-proposal)# authentication algorithm md5
```

authentication mode

This command sets the XAUTH authentication mode of remote users connecting via IPsec.

The use of a negative form (no) of the command removes a set mode.

Syntax

```
authentication mode { local | radius | client }
no authentication mode
```

Parameters

local - authentication mode using the local user base of the configured profile;

radius - the mode in which user authentication passes through a RADIUS server;

client – mode used by the xauth client.

Required privilege level

15

Command mode

CONFIG-IKE-POLICY

Example

```
esr(config-ike-policy)# authentication mode local
```

authentication method

This command selects the key authentication method for the IKE connection. Message authentication by key is used when an IKE connection is established, the key is set in the IKE policy (see section pre-shared-key). After an IKE connection is established, message authentication is performed using a hashing algorithm.

The use of a negative form (no) of the command sets the default value.

Syntax

```
authentication method <METHOD>
no authentication method
```

Parameters

<METHOD> – key authentication method. May take the following values:

- pre-shared-key – authentication method using pre-received encryption keys;
- rsa-public-key – authentication method using RSA certificate;
- xauth-psk-key – an extended authentication method using local or remote user databases.

Default value

pre-shared-key

Required privilege level

15

Command mode

CONFIG-IKE-POLICY

Example

```
esr(config-ike-proposal)# authentication method pre-shared-key
```

bind-interface vti

This command specifies the tunnel interface through which traffic will pass in the 'route-based' tunnel mode.

The use of a negative form (no) of the command removes a bind to tunnel interface.

Syntax

```
bind-interface vti <VTI>
no bind-interface vti
```

Parameters

<VTI> – VTI ID.

Required privilege level

10

Command mode

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-gw)# bind-interface vti 1
```

certificate

This command specifies the necessary certificates.

The use of a negative form (no) of the command removes certificate name from the configuration.

Syntax

```
certificate <CERTIFICATE-TYPE> <NAME>
no certificate <CERTIFICATE-TYPE>
```

Parameters

<CERTIFICATE-TYPE> – certificate or key type, may take the following values:

- ca – certificate authority certificate;
- crl – certificate revocation list;
- local-crt – local side certificate;
- local-crt-key – RSA key of the local side certificate;
- local-id – local side ID. The key «any» is used to disable the verification of the Subject attribute fields of the local certificate;
- remote-crt – remote side certificate. Instead of the file name, it is possible to use the key «any» to set the mode of reception of the public key of the remote party over the network;
- remote-id – remote side ID. The key «any» is used to disable the verification of the Subject attribute fields of the remote party certificate;

<NAME> – certificate or key name, set by the string of up to 31 characters.

Default value

None

Required privilege level

15

Command mode

CONFIG-IKE-POLICY

Example

```
esr(config-ike-policy)# certificate ca KEY
```

data-tunnel address

This command specifies the IP address for building a GRE data tunnel that is sent to the client, connected via IPsec using dynamic parameter configuration. GRE data tunnel must be supported on the client side. (Requires ELTEX_DATA_IP(28684).

The use of a negative form (no) of the command removes the IP address for GRE data tunnel building.

Syntax

```
data-tunnel address <ADDR>
no data-tunnel address
```

Parameters

<ADDR> – IP address for GRE data tunnel building, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Default value

None

Required privilege level

10

Command mode

CONFIG-POOL

Example

```
esr(config-pool)# data-tunnel address 192.168.2.66
```

dead-peer-detection action

This command sets the action that the device should take, in case of detection of unavailability of IPSec neighbor, by the mechanism of Dead Peer Detection.

Dead Peer Detection (DPD) is a mechanism for checking the status and availability of neighboring devices. The mechanism periodically sends R-U-THERE messages (for IKE version 1) or empty INFORMATIONAL messages (for IKE version 2) to check the availability of the IPsec neighbor.

The use of a negative form (no) of the command sets the default value.

Syntax

```
dead-peer-detection action <MODE>
```

```
no dead-peer-detection action
```

Parameters

<MODE> – DPD operation mode:

- restart – connection restarts;
- clear – conection stops;
- hold – connection holds;
- none - the mechanism is disabled, no action is taken.

Default value

none

Required privilege level

10

Command mode

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-gw)# dead-peer-detection action clear
```

dead-peer-detection interval

This command sets the interval between sending messages by the DPD mechanism.

The DPD mechanism is described in section certificate.

The use of a negative form (no) of the command sets the default value.

Syntax

```
dead-peer-detection interval <SEC>
no dead-peer-detection interval
```

Parameters

<SEC> – interval between sending messages via DPD mechanism, takes values of [1..180] seconds.

Default value

2 seconds

Required privilege level

10

Command mode

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-gw)# dead-peer-detection interval 15
```

dead-peer-detection timeout

This command sets the response timeout for messages sent by the DPD mechanism.

The DPD mechanism is described in section certificate.

The use of a negative form (no) of the command sets the default value.

Syntax

```
dead-peer-detection timeout <SEC>
no dead-peer-detection timeout
```

Parameters

<SEC> – time interval of response to DPD mechanism messages, takes values of [1..180] seconds.

Default value

30 seconds

Required privilege level

10

Command mode

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-gw)# dead-peer-detection timeout 60
```

description

The command changes the profile, policy, or gateway to the IKE protocol description.

The use of a negative form (no) of the command removes description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – profile description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-IKE-PROPOSAL

CONFIG-IKE-POLICY

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-proposal)# description "my proposal"
```

dh-group

This command sets the group number of the Diffie-Hellman method. The group number defines the level of security of the IKE connection when exchanging keys — security increases as the group number increases, but the connection establishment time increases.

The use of a negative form (no) of the command sets the default value.

Syntax

```
dh-group <DH-GROUP>
```

```
no dh-group
```

Parameters

<DH-GROUP> – Diffie-Hellman group number, takes values of [1, 2, 5, 14, 15, 16, 17, 18].

Default value

1

Required privilege level

15

Command mode

CONFIG-IKE-PROPOSAL

Example

```
esr(config-ike-proposal)# dh-group 5
```

encryption algorithm

This command selects the encryption algorithm used when establishing an IKE connection.

The use of a negative form (no) of the command sets the default value.

Syntax

```
encryption algorithm <ALGORITHM>
```

```
no encryption algorithm
```

Parameters

<ALGORITHM> – encryption protocol ID, takes the following values: des, 3des, blowfis28, blowfis92, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.

Default value

3des

Required privilege level

15

Command mode

CONFIG-IKE-PROPOSAL

Example

```
esr(config-ike-proposal)# encryption algorithm aes128
```

ike-policy

This command establishes the binding of the IKE protocol policy to the gateway.

The use of a negative form (no) of the command removes poilitics binding.

Syntax

```
[no] ike-policy <NAME>
```

Parameters

<NAME> – IKE protocol policy name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-gw)# ike-policy ike_pol1
```

ip prefix

This command specifies the address pool from which addresses will be issued to IPsec clients.

The use of a negative form (no) of the command removes the address pool from which addresses will be issued to IPsec clients.

Syntax

```
ip prefix <ADDR/LEN>
no ip prefix
```

Parameters

<ADDR/LEN> – IP subnet, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32].

Default value

Unspecified.

Required privilege level

10

Command mode

CONFIG-POOL

Example

```
esr(config-pool)# ip prefix 192.168.0.0/16
```

lifetime seconds

This command sets the lifetime of the IKE protocol connection.

The use of a negative form (no) of the command sets the default value.

Syntax

```
lifetime seconds <SEC>
no lifetime seconds
```

Parameters

<SEC> – time interval, takes values of [4..86400] seconds.

Default value

10800 seconds

Required privilege level

10

Command mode

CONFIG-IKE-POLICY

Example

```
esr(config-ike-proposal)# lifetime 21600
```

local address

The command sets IP address of a local IPsec tunnel gateway.

The use of a negative form (no) of the command removes local gateway IP address.

Syntax

`local address <ADDR>`

`no local address`

Parameters

<ADDR> – IP address of a local gateway.

Required privilege level

10

Command mode

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-gw)# local address 192.168.1.1
```

local interface

The command sets the use of IP address assigned to the interface as a IPsec tunnel local gateway.

The use of a negative form (no) of the command stops the use of IP address assigned to the interface as a local gateway.

Syntax

`local interface <IF>`

`no local interface`

Parameters

<IF> – interface type and identifier specified in the form described in Section Types and naming order of router interfaces.

Required privilege level

10

Command mode

CONFIG-IKE-GW

Example

```
esr(config-ike-gw)# local interface gigabitethernet 1/0/1
```

local network

This command sets sender's subnet IP address as well as IP and port. Traffic that meets the specified criteria will be sent to the IPsec tunnel.

The use of a negative form (no) of the command removes senders subnet IP address.

Syntax

```
[no] local network <ADDR/LEN> [ protocol { <TYPE> | <ID> } [ port <PORT> ] ]
```

Parameters

<ADDR/LEN> – IP subnet of a sender. The parameter is defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32];

<TYPE> – protocol type, takes the following values: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ID> – IP identification number, takes values of [0x00-0xFF];

<PORT> – TCP/UDP port, takes values of [1..65535].

Required privilege level

10

Command mode

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-gw)# local network 192.168.1.0/24 protocol tcp port 22
```

management-tunnel address

This command specifies the tunnel IP address for building a GRE management tunnel that is sent to the client, connected via IPsec using dynamic parameter configuration. GRE management tunnel must be supported on the client side. (Requires ELTEX_MANAGEMENT_IP(28683).

The use of a negative form (no) of the command removes the tunnel IP address for GRE management tunnel building.

Syntax

```
management-tunnel address <ADDR>
no management-tunnel address
```

Parameters

<ADDR> – IP address for GRE management tunnel building, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Default value

None

Required privilege level

10

Command mode

CONFIG-POOL

Example

```
esr(config-pool)# management-tunnel address 192.168.2.87
```

mobike disable

This command disables the ability of the connection initiator to change its network connection point (to use the IP address as the local address parameter).

The use of a negative form (no) of the command activates the automatic selection of local addres when the described in the configuration is not available.

Syntax

```
[ no ] mobike disable
```

Parameters

None.

Default value

Enabled.

Required privilege level

10

Command mode

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-gateway)# mobike disable
```

mode

This command sets the negotiation mode for the first phase of the IKE protocol.

The use of a negative form (no) of the command sets the default value.

Syntax

`mode <MODE>`

`no mode`

Parameters

<MODE> – first IKE phase mode, may take values:

- main – consists of three bilateral exchanges between the sender and the recipient:
  - During the first exchange, the authentication and encryption algorithms that will be used to protect the IKE connection by matching the IKE protocol profiles of each node will be matched;
  - Using the Diffie-Hellman algorithm, the parties exchange a common secret key. The nodes also check each other's identification by sending and confirming a sequence of pseudo-random numbers;
  - The identity of the opposite side is checked. As a result of the main mode execution, a secure channel is created for the second phase of the IKE protocol.
- aggressive – this mode is costing fewer exchanges and, accordingly, the number of packets:
  - The first message (from the initiator) sends information that is used to establish an IKE connection: a suggestion of the SA parameters, the initiation of a Diffie-Hellman exchange, the sending of a pseudo-random number and a packet identifier;
  - In the second message, the responder accepts the SA, authenticates the initiator, sends a pseudo-random number and its IKE-identifier;
  - In the third message, the initiator authenticates the responder and confirms the exchange.

Default value

main

Required privilege level

15

Command mode

CONFIG-IKE-POLICY

Example

```
esr(config-ike-policy)# mode aggressive
```

mode

This command sets the mode of traffic redirection to the tunnel.

The use of a negative form (no) of the command sets the default value.

Syntax

`mode <MODE>`

`no mode`

Parameters

<MODE> – mode of traffic redirection into the tunnel, takes the following values:

- policy-based — traffic is redirected based on the subnets specified in the policies;
- route-based — traffic is redirected based on routes whose gateway is a tunnel interface.

Required privilege level

10

Command mode

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-gw)# mode route-based
```

password

This command is used to set the user password for IKE-GETWAY. The password can be set both in clear text and in the form of sha512 hash.

The use of a negative form (no) of the command removes user's password for IKE-GETWAY from the system.

Syntax

```
password ascii-text { <CLEAR-TEXT> | encrypted <HASH_SHA512> }
no password
```

Parameters

<CLEAR-TEXT> – password, set by the string of 8 to 32 characters, takes the value of [0-9a-fA-F].

<HASH_SHA512> – hash password via sha512 algorithm, set by the string of 110 characters;

Required privilege level

15

Command mode

CONFIG-PROFILE

Example

```
esr(config-profile) password tteesstt
```

password local-crt-key

This command is used to set the password from the encrypted certificate chain (certificates are assigned using the certificate command)

The use of a negative form (no) of the command removes the password.

Syntax

```
password local-crt-key ascii-text { <CLEAR-TEXT> | encrypted <HASH_SHA512> }
no password local-crt-key
```

Parameters

<CLEAR-TEXT> – password, set by the string of 8 to 32 characters, takes the value of [0-9a-fA-F].

<HASH_SHA512> – hash password via sha512 algorithm, set by the string of 110 characters;

Required privilege level

15

Command mode

CONFIG-IKE-POLICY

Example

```
esr(config-ike-policy) password tteesstt
```

pfs dh-group

This command sets the group number of the Diffie-Hellman method. The group number defines the level of security of the IPsec connection when exchanging keys — security increases as the group number increases, but the connection establishment time increases.

The use of a negative form (no) of the command sets the default value.

Syntax

```
pfs dh-group <DH-GROUP>
```

```
no pfs dh-group
```

Parameters

<DH-GROUP> – Diffie-Hellman group number, takes values of [1, 2, 5, 14, 15, 16, 17, 18].

Default value

1

Required privilege level

15

Command mode

CONFIG-IPSEC-PROPOSAL

Example

```
esr(config-isec-proposal)# pfs dh-group 5
```

pre-shared-key

This command specifies a shared secret authentication key that should be the same for both parties of the tunnel.

The use of a negative form (no) of the command removes a set key.

Syntax

```
pre-shared-key { ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> } | hexadecimal { <HEX>
| encrypted <ENCRYPTED-HEX> } }
```

```
no pre-shared-key
```

Parameters

<TEXT> – string [1..64] ASCII characters;

<HEX> – number, [1..32] bytes size, set by the string of [2..128] characters in hexadecimal format (0xYYYY ...) or (YYYY ...).

<ENCRYPTED-TEXT> – encrypted password, [1..32] bytes size, set by the string of [2..128] characters.

<ENCRYPTED-TEXT> – encrypted number, [2..64] bytes size, set by the string of [2..256] characters.

Default value

none

Required privilege level

15

Command mode

CONFIG-IKE-POLICY

Example

```
esr(config-ike-policy)# pre-shared-key hexadecimal abc123
```

proposal

This command establishes the binding of the IKE protocol profile to the policy.

The use of a negative form (no) of the command removes IKE protocol profile binding.

Syntax

`[no] proposal <NAME>`

Parameters

<NAME> – IKE protocol name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-IKE-POLICY

Example

```
esr(config-ike-policy)# proposal ike_prop1
```

remote address

The command sets IP address of a remote IPsec tunnel gateway.

The use of a negative form (no) of the command removes remote gateway IP address.

Syntax

```
remote address { <ADDR> | any }
```

```
no remote address
```

Parameters

<ADDR> – IP address of a remote gateway.

any – key that allows you to receive requests to establish an IKE session from any IP address.

Required privilege level

10

Command mode

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-gw)# remote address 192.168.1.2
```

remote network

This command sets the IP address of the receiver's subnet, as well as the IP protocol and port, or assigns a dynamic address pool for remote clients using XAUTH. Traffic that meets the specified criteria will be sent to the IPsec tunnel.

The use of a negative form (no) of the command removes senders subnet IP address.

Syntax

```
remote network { dynamic pool <POOL> | <ADDR/LEN> [ protocol { <TYPE> | <ID> } [ port <PORT> ] ] | any }
```

```
no remote network { dynamic pool |<ADDR/LEN> [ protocol { <TYPE> | <ID> } [ port <PORT> ] ] | any }
```

Parameters

<POOL> – dedicated dynamic address pool for XAUTH clients;

<ADDR/LEN> – IP subnet of a recipient. The parameter is defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32];

<TYPE> – protocol type, takes the following values: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ID> – IP identification number, takes values of [0x00-0xFF];

<PORT> – TCP/UDP port, takes values of [1..65535];

any – key indicating the need to encrypt any outgoing traffic.

Required privilege level

10

Command mode

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-gw)# remote network 192.168.0.0/24 protocol tcp port 22
```

remote network dynamic client

This command enables receiving a list of remote networks from an IPsec-VPN server.

The use of a negative form of the command (no) disables the receiption of a list of remote networks from the IPsec-VPN server.

Syntax

```
[no] remote network dynamic client
```

Parameters

None.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-IKE-GW

Example

```
esr(config-ike-gw)# remote network dynamic client
```

security ike gateway

This command switches to the command configuration mode of the IKE SECURITY IKE GATEWAY gateway. If an IKE gateway with the specified name does not exist in the configuration, it will be created. Gateway parameters include the VTI interface to which the traffic will be sent, the policy and version of the IKE protocol, and also the mode of forwarding traffic to the tunnel.

The use of a negative form (no) of the command removes IKE protocol gateway.

Syntax

```
[no] security ike gateway <NAME>
```

Parameters

<NAME> – IKE protocol gateway name, set by the string of up to 31 characters. The use of a negative form (no) of the command with ‘all’ parameter removes all IKE gateways.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# security ike gateway ike_gw1
esr(config-ike-gw)#
```

security ike policy

This command creates an IKE policy that includes IKE protocol profiles, a shared secret key for authentication, and a negotiation mode for the first phase of the IKE protocol.

The use of a negative form (no) of the command removes a specified policy. The command sets the command line mode to SECURITY IKE POLICY.

Syntax

```
[no] security ike policy <NAME>
```

Parameters

<NAME> – IKE policy name, set by the string of up to 31 characters. The use of a negative form (no) of the command with ‘all’ parameter removes all IKE policy.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# security ike policy ike_pol1
esr(config-ike-policy)#
```

security ike proposal

This command creates an Internet Key Exchange (IKE) protocol profile that includes the encryption and authentication parameters of the Diffie-Hellman method, which will be used when negotiating IKE parameters with the opposite side of the VPN connection when creating the Security Association (SA). In addition, the profile sets the SA limit time. The use of a negative form (no) of the command removes a specified profile.

Syntax

`[no] security ike proposal <NAME>`

Parameters

<NAME> – IKE protocol name, set by the string of up to 31 characters. The use of a negative form (no) of the command with 'all' parameter removes all IKE profiles.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# security ike proposal ike_prop1
esr(config-ike-proposal)#
```

security ike session uniqueids

This command sets the mode for reconnecting XAUTH clients with one login/password.

The use of a negative form (no) of the command sets the default value.

Syntax

```
security ike session uniqueids <MODE>
no security ike session uniqueids
```

Parameters

<MODE> – reconnect mode, may take the following values:

- no – established XAUTH connection will be deleted if an «INITIAL_CONTACT» notification is sent for a new XAUTH connection by the initiator of the connection, the previously used IP address will be assigned. Otherwise, the established XAUTH connection will be withheld. A new IP address will be assigned to the new XAUTH connection.
- never – established XAUTH connection will be withheld. A new IP address will be assigned to the new XAUTH connection. The «INITIAL_CONTACT» notification will be ignored anyway.
- replace – established XAUTH connection will be deleted. The previously used IP address will be used for the new XAUTH connection.
- keep – established XAUTH connection will be withheld. A new XAUTH connection will be rejected.

Default value

never

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# security ike session uniqueids replace
```

show security ike

The command is used to view a list of gateways, policies or profiles.

Syntax

```
show security ike { gateway | policy | proposal } [<NAME>]
```

Parameters

gateway - if the 'gateway' command is specified, the list of configured gateways will be displayed;

policy - when specifying the 'policy' command, a list of configured policies will be displayed;

proposal - if you specify the 'proposal' command, a list of configured profiles will be displayed;

<NAME> – name. If you specify a specific gateway name, policy, profile, detailed information will be displayed.

Required privilege level

10

Command mode

ROOT

Example

```
esr# show security ike proposal
   Proposal
   ~~~~~~~~
Name           Auth      Encryption         DH   Hash         Lifetime
------------   -------   ----------------   --   ----------   ----------
aaa            pre-sha   3des               1    sha1         3600
               red-key
 esr# show security ike policy
   Policy
   ~~~~~~
Name                           Mode         Proposal
----------------------------   ----------   -----------------------------------
ike_pol1                       main         ike_prop1
 esr# show security ike gateway ik_gw
Description:                --
IKE Policy:                 ike_pol1
IKE Version:                v1-only
Mode:                       route-based
Binding interface:          vti1
IKE Dead Peer Detection:
    Action:                 none
    Interval:               2
    Timeout:                30
```

user

This command sets the username for IKE-GATEWAY authentication.

The use of a negative form (no) of the command removes a specified user.

After executing this command, the router enters the user password configuration mode (config-profile).

Syntax

```
[no] user <NAME>
```

Parameters

<NAME> – user name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-ACCESS-PROFILE

Example

```
esr(config-access-profile)# user connecter963
```

version

This command sets the IKE protocol version.

The use of a negative form (no) of the command sets the default value.

Syntax

```
version <VERSION>
no version
```

Parameters

<version> – IKE protocol version: v1-only or v2-only.

Default value

v1-only

Required privilege level

15

Command mode

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-gw)# version v2-only
```

xauth access-profile

This command specifies the local list of users for authorization XAUTH.

The use of a negative form (no) of the command removes a specified profile.

Syntax

```
[no] xauth access-profile <NAME> [client <USER-NAME>
```

Parameters

<NAME> – local XAUTH user list name, set by the string of up to 31 characters;

<USER-NAME> – username from the attached xauth-profile is specified by a string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-IKE-GATEWAY

Example

```
esr(config-ike-gateway)# xauth access-profile OFFICE
```

VPN management. IPsec configuration

authentication algorithm

The command sets an authentication algorithm. The use of a negative form (no) of the command sets the default value.

Syntax

```
authentication algorithm <ALGORITHM>
no authentication algorithm
```

Parameters

<ALGORITHM> – authentication algorithm, takes values of: md5, sha1, sha2-256, sha2_384, sha2-512.

Default value

sha1

Required privilege level

15

Command mode

CONFIG-IPSEC-PROPOSAL

Example

```
esr(config-ipsec-proposal)# authentication algorithm md5
```

description

This command changes the description.

The use of a negative form (no) of the command removes description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – profile description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-IPSEC-VPN

CONFIG-IPSEC-PROPOSAL

CONFIG-IPSEC-POLICY

Example

```
esr(config-ipsec-vpn)# description "VPN to Moscow Office"
```

enable

This command enables IPsec VPN.

The use of a negative form of the command (no) disables IPsec VPN.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

Disabled

Required privilege level

10

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# enable
```

encryption algorithm

The command sets encryption algorithm. The use of a negative form (no) of the command sets the default value.

Syntax

`encryption algorithm <ALGORITHM>`

`no encryption algorithm`

Parameters

<ALGORITHM> – encryption protocol, takes the following values: null, des, 3des, blowfis28, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.

Default value

3des

Required privilege level

15

Command mode

CONFIG-IPSEC-PROPOSAL

Example

```
esr(config-ipsec-proposal)# encryption algorithm blowfish128
```

ike dscp

The command sets the DSCP code value for the use in IP headers of IKE protocol outgoing packets.

The use of a negative form (no) of the command sets the default DSCP value.

Syntax

```
ike dscp <DSCP>
no ike dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Default value

63

Required privilege level

10

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# ike dscp 40
```

ike establish-tunnel

This command sets VPN activation mode. This command is relevant only if the 'ike' key agreement mode is selected in VPN. Key agreement mode configuration is described in mode.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ike establish-tunnel <MODE>
no Ike establish-tunnel
```

Parameters

<MODE> – VPN activation mode:

- by-request – connection is enabled by an opposing party;
- route – connection is enabled when there is traffic routed to the tunnel;
- immediate – tunnel is enabled automatically after applying the configuration.

Required privilege level

15

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# ike establish-tunnel route
```

ike gateway

This command binds the IKE gateway to the VPN. This command is relevant only if the 'ike' key agreement mode is selected in VPN. Key agreement mode configuration is described in mode.

Syntax

`ike gateway <NAME>`

`no ike gateway`

Parameters

<NAME> – IKE gateway name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# ike gateway ike_gw1
```

ike idle-time

This command sets the time interval value in seconds after which the connection is closed, if no packet has been received or sent via SA (optionally)

The use of a negative form (no) of the command disables this timer.

Syntax

`ike idle-time <TIME>`

`no ike idle-time`

Parameters

<TIME> – interval in seconds, takes values of [4..86400].

Required privilege level

10

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# ike idle-time 3600
```

ike rekey disable

Disable key re-approval before the IKE connection is lost due to the timeout, the number of transmitted packets or bytes.

The use of a negative form (no) of the command enables the renegotiation of keys.

Syntax

```
[no] ike rekey disable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# ike rekey disable
```

ike rekey margin

This command allows you to configure the start of the renegotiation of the keys of an IKE connection before the expiration of the lifetime.

The use of a negative form (no) of the command sets the default value.

Syntax

```
Ike rekey margin { seconds <SEC> | packets <PACKETS> | kilobytes <KB> }
```

```
no ike rekey margin { seconds | packets | kilobytes }
```

Parameters

<SEC> – time interval in seconds remaining before the connection release (set by the lifetime seconds command, see lifetime). Takes values in the range of [4..86400].

<PACKETS> – number of packets remaining before the connection release (set by the lifetime packets command, see lifetime). Takes values in the range of [4..86400].

<KB> – traffic volume in kilobytes remaining before the connection release (set by the lifetime kilobytes command, see lifetime). Takes values in the range of [4..86400].

Default value

- Keys re-approval before the expire of time – 540 seconds before.

- Keys re-approval before the expire of traffic volume and amount of packets – disabled.

Required privilege level

15

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# ike rekey margin seconds 1800
```

ike rekey randomization

This command sets the level of margin seconds, margin packets, margin kilobytes values random spread (optionally).

The use of a negative form (no) of the command sets the default value.

Syntax

```
ike rekey randomization <VALUE>
```

```
no ike rekey randomization
```

Parameters

<VALUE> – maximum ratio of values spread, takes values of [1..100].

Default value

100%

Required privilege level

15

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# ike rekey randomization 10
```

ike ipsec-policy

This command associates the IPsec policy with the VPN. This command is relevant only if the 'ike' key agreement mode is selected in VPN. Key agreement mode configuration is described in mode.

Syntax

```
ike ipsec-policy <NAME>
no ike ipsec-policy
```

Parameters

<NAME> – IPsec policy name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# ike ipsec-policy ipsec_pol1
```

lifetime

This command sets the lifetime of the IPsec tunnel.

The use of a negative form (no) of the command sets the default value.

Syntax

```
lifetime { seconds <SEC> | packets <PACKETS> | kilobytes <KB> }
no lifetime { seconds | packets | kilobytes }
```

Parameters

<SEC> – IPsec tunnel lifetime after which the re-approval is carried out. Takes values in the range of [1140..86400] seconds.

<PACKETS> – number of packets after transmitting of which the IPsec tunnel re-approval is carried out. Takes values in the range of [4..86400].

<KB> – traffic amount after transmitting of which the IPsec tunnel re-approval is carried out. Takes values in the range of [4..86400] seconds.

Default value

3600 seconds

Required privilege level

10

Command mode

CONFIG-IPSEC-POLICY

Example

```
esr(config-ipsec-proposal)# lifetime seconds 3600
```

manual authentication algorithm

The command sets an authentication algorithm. This command is relevant only if the 'manual' key matching mode is selected in VPN. Key agreement mode configuration is described in mode.

The use of a negative form (no) of the command sets the default value.

Syntax

```
manual authentication algorithm <ALGORITHM>
no manual authentication algorithm
```

Parameters

<ALGORITHM> – authentication algorithm, takes values of: md5, md5-128, sha1, sha1-160, aesxcbc, sha2-256, sha2-384, sha2-512].

Default value

none

Required privilege level

15

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# manual authentication algorithm sha1
```

manual authentication key

The command sets an authentication key. This command is relevant only if the 'manual' key agreement mode is selected in VPN. Key agreement mode configuration is described in mode.

Syntax

```
manual authentication key { ascii-text {<TEXT> | encrypted <ENCRYPTED-TEXT>} |
hexadecimal {<HEX> | encrypted <ENCRYPTED-HEX> } }
```

```
no manual authentication key
```

Parameters

<TEXT> – string [1..64] ASCII characters;

<HEX> – number, [1..32] bytes size, set by the string of [2..128] characters in hexadecimal format (0xYYYY ...) or (YYYY ...);

<ENCRYPTED_TEXT> – encrypted password, [1..32] bytes size, set by the string of [2..128] characters.

<ENCRYPTED_HEX> – encrypted number, [2..64] bytes size, set by the string of [2..256] characters.

Required privilege level

15

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# manual authentication key hexadecimal abcdef
```

manual bind-interface vti

This command specifies the tunnel interface through which traffic will pass in the 'route-based' tunnel mode. This command is relevant only if the 'manual' key agreement mode is selected in VPN. Key agreement mode configuration is described in mode.

The use of a negative form (no) of the command sets the default value.

Syntax

```
manual bind-interface vti <VTI>
no manual bind-interface vti
```

Parameters

<VTI> – VTI interface index, takes the values:

ESR-10/12V/12VF/14VF – [1..10];

ESR-20/21/100/200 – [1..250];

ESR-1000/1200/1500/1511/1700/3100 – [1..500].

Required privilege level

10

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# manual bind-interface vti 0
```

manual encryption algorithm

The command sets encryption algorithm. This command is relevant only if the 'manual' key agreement mode is selected in VPN. Key agreement mode configuration is described in mode.

The use of a negative form (no) of the command removes a specified value.

Syntax

```
manual encryption algorithm <ALGORITHM>
no manual encryption algorithm
```

Parameters

<ALGORITHM> – encryption algorithm, takes the following values: des, 3des, blowfis28, blowfis92, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.

Default value

3des

Required privilege level

15

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# manual encryption algorithm blowfis28
```

manual encryption key

The command sets encryption key. This command is relevant only if the 'manual' key agreement mode is selected in VPN. Key agreement mode configuration is described in mode.

The use of a negative form (no) of the command removes a specified value.

Syntax

```
manual encryption key { ascii-text { < TEXT> | encrypted <ENCRYPTED-TEXT> } | hexadecimal { <HEX> | encrypted <ENCRYPTED-HEX> } }
```

```
no manual encryption key
```

Parameters

<TEXT> – string [1..36] ASCII characters;

<HEX> – number, [1..24] bytes size, set by the string of [2..72] characters in hexadecimal format (0xYYYY ...) or (YYYY ...);

<ENCRYPTED-TEXT> – encrypted password, [1..24] bytes size, set by the string of [2..72] characters;

<ENCRYPTED-HEX> – encrypted number, [2..36] bytes size, set by the string of [2..144] characters.

Required privilege level

15

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# manual encryption key hexadecimal 0x123456
```

manual mode

This command sets the mode of traffic redirection to the tunnel. This command is relevant only if the 'manual' key agreement mode is selected in VPN. Key agreement mode configuration is described in mode.

The use of a negative form (no) of the command sets the default value.

Syntax

```
manual mode <MODE>
```

```
no manual mode
```

Parameters

<MODE> - traffic passing mode:

- policy-based — traffic is redirected based on the subnets specified in the policies;
- route-based — traffic is redirected based on routes whose gateway is a tunnel interface.

Required privilege level

10

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# manual mode route-based
```

manual protocol

The command sets encapsulation protocol. This command is relevant only if the 'manual' key agreement mode is selected in VPN. Key agreement mode configuration is described in mode.

The use of a negative form (no) of the command sets the default value.

Syntax

```
manual protocol <TYPE>
```

```
no manual protocol
```

Parameters

<TYPE> – protocol type, takes the following values:

- ah - this protocol performs only traffic authentication, data encryption is not performed;
- esp - this protocol authenticates and encrypts traffic.

Default value

esp

Required privilege level

15

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# manual protocol ah
```

manual spi

This command sets the index of security settings. This command is relevant only if the 'manual' key agreement mode is selected in VPN. Key agreement mode configuration is described in mode.

The use of a negative form (no) of the command removes a specified security parameters index.

Syntax

`manual spi <HEX>`

`no manual spi`

Parameters

<HEX> – an index of security parameters, set to 32 bits (8 characters) in hexadecimal format (0xYYYY ...) or (YYYY ...).

Required privilege level

15

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# manual spi FF
```

mode

This command defines the matching mode of data required for VPN activation.

Syntax

```
mode <MODE>
```

```
no mode
```

Parameters

<MODE> – VPN operation mode:

- ike – coordination of authentication and encryption algorithms, authentication and encryption keys, security parameter index and other data is carried out through the IKE protocol;
- manual - the user must configure identical parameters on both nodes for the VPN to work. This mode does not establish an IKE connection between nodes. Each node encrypts and decrypts packets based only on the specified parameters.

Required privilege level

15

Command mode

CONFIG-IPSEC-VPN

Example

```
esr(config-ipsec-vpn)# mode ike
```

proposal

This command binds IPsec protocol set profiles to the policy.

The use of a negative form (no) of the command removes a bind to a specified profile.

Syntax

```
[no] proposal <NAME>
```

Parameters

<NAME> – IPsec protocol set profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-IPSEC-POLICY

Example

```
esr(config-ipsec-policy)# proposal ipsec_prop1
```

protocol

The command sets encapsulation protocol.

The use of a negative form (no) of the command sets the default value.

Syntax

```
protocol <PROTOCOL>
no protocol
```

Parameters

<PROTOCOL> – encapsulation protocol, takes the following values:

- ah - this protocol performs only traffic authentication, data encryption is not performed;
- esp - this protocol authenticates and encrypts traffic.

Default value

esp

Required privilege level

15

Command mode

CONFIG-IPSEC-PROPOSAL

Example

```
esr(config-ipsec-proposal)# protocol ah
```

security ipsec policy

This command creates an IPsec protocol dial policy that includes IPsec protocol suite profiles for negotiating the second phase of the IKE protocol.

The use of a negative form (no) of the command removes a specified value.

The command sets the command line mode to SECURITY IPSEC POLICY.

Syntax

`[no] security ipsec policy <NAME>`

Parameters

<NAME> – IPsec policy name, set by the string of up to 31 characters. The use of a negative form (no) of the command with ‘all’ parameter removes all IPsec policy.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# security ipsec policy ipsec_pol1
esr(config-ipsec-policy)#
```

security ipsec proposal

This command creates a profile for the IPsec protocol suite. The IPsec profile includes the parameters of the encryption and authentication algorithms, the security protocol of the IPsec tunnel connection, and the lifetime of the connection.

The use of a negative form (no) of the command removes a specified profile.

The command sets the command line mode to SECURITY IPSEC PROPOSAL.

Syntax

`[no] security ipsec proposal <NAME>`

Parameters

<NAME> – IPsec profile name, set by the string of up to 31 characters. The use of a negative form (no) of the command with 'all' parameter removes all IPsec profiles.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# security ipsec proposal ipsec_prop1
esr(config-ipsec-proposal)#
```

security ipsec vpn

This command creates a VPN based on the IPsec protocol suite and sets the SECURITY IPSEC VPN command mode.

The use of a negative form (no) of the command removes a configured VPN.

Syntax

`[no] security ipsec vpn <NAME>`

Parameters

<NAME> – VPN name, set by the string of up to 31 characters. The use of a negative form (no) of the command with ‘all’ parameter removes all VPN.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# security ipsec vpn ipsec_vpn1
esr(config-ipsec-vpn)#
```

show security ipsec

This command displays the VPN configurations, policies, and IPsec protocol suite profiles.

Syntax

`show security ipsec { vpn configuration | policy | proposal } [<NAME>]`

Parameters

vpn configuration – if this command is specified, the configuration of all VPNs will be displayed;

vpn status – if this command is specified, the operational status of all VPNs will be displayed;

policy — specifying this command will display a list of configured IPsec protocol set policies;

proposal - specifying this command will display a list of configured IPsec protocol set profiles;

<NAME> – name. If you specify a specific name of VPN, policy, or profile detailed information will be displayed.

Required privilege level

10

Command mode

ROOT

Example

```
esr# show security ipsec proposal
   Proposal
   ~~~~~~~~
Name                    Prot   Enc. alg.          Auth. alg.        Lifetime
---------------------   ----   ----------------   ---------------   -----------
ipsec_prop1             esp    aes128             sha1              28800 sec
esr# show secu  rity ipsec policy
Name                   Description           Proposal
--------------------   -------------------   -----------------------------------
ipsec_pol1                                   ipsec_prop1
Master# show security ipsec vpn configuration IPSECVPN
Description:                --
State:                      Enabled
IKE:
    Establish tunnel:           immediate
    IPsec policy:               IPSECPOLICY
    IKE gateway:                IKEGW
    IKE DSCP:                   63
    IKE idle-time:              0s
    IKE rekeying:               Enabled
        Margin time:                540s
        Margin kilobytes:           0
        Margin packets:             0
        Randomization:              100%
```

show security ipsec vpn authentication

This command allows you to see the list and parameters of the connected IPsec-VPN clients.

Syntax

`show security ipsec vpn authentication <NAME> [ vrf <VRF> ]`

Parameters

<NAME> – IPsec VPN name, set by the string of up to 31 characters.

<VRF> – VRF instance name, set by the string of up to 31 characters, within which DNS names resolution will be enabled.

Required privilege level

10

Command mode

ROOT

Example

```
esr# show security ipsec vpn authentication
Local host        Remote host        Local subnet          Remote subnet         Authentication
State
---------------   ---------------   --------------------   --------------------
-----------------------------------------   -----------
2.2.2.1           2.2.2.2           192.168.2.0/24         192.168.1.1/32        Xauth PSK,
login: ipsec                     Established
```

show security ipsec vpn status

This command shows the status of all VPNs that establish a connection through the IKE protocol or a specific VPN when specifying its name.

Syntax

```
show security ipsec vpn status [ vrf <VRF> ] [ <NAME> ]
```

Parameters

<NAME> – VPN name, set by the string of up to 31 characters.

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

ROOT

Example

```
esr# show security ipsec vpn status
Name      Local host   Remote host  Initiator spi       Responder spi       State
--------- ------------ ------------ ---------------     ---------------     ------
ipsec_vpn1 10.100.14.1 10.100.14.2  0x05d8e0ac3543f0cb  0xcfa1c4179d001154   Established
```

22 VPN management. Remote access settings

- General commands for remote access configuration
  - clear remote-access counters
  - clear remote-access session
  - description
  - enable
  - encryption mppe
  - remote-access
  - show remote-access configuration
  - show remote-access counters
  - show remote-access status
- L2TP over IPsec/PPTP server configuration
  - authentication mode
  - authentication method
  - dns-servers
  - dscp
  - ipsec authentication method
  - ipsec authentication pre-shared-key
  - local-address
  - mtu
  - outside-address
  - remote-address
  - remote network
  - remote networks
  - username
  - wins-servers
- OpenVPN server configuration
  - address-range
  - authentication algorithm
  - bridge-group
  - certificate
  - client-isolation
  - client-max
  - compression
  - dns-server
  - duplicate-cn
  - encryption algorithm
  - ip address
  - login authentication
  - network
  - port
  - protocol
  - redirect-gateway
  - route
  - timers holdtime
  - timers keepalive
  - subnet
  - tunnel
  - username
  - wins-server

General commands for remote access configuration

clear remote-access counters

This command resets the connection counters for OpenVPN, PPTP and L2TP over IPsec users.

Syntax

```
clear remote-access counters [ pptp | l2tp | openvpn ] [ server <SERVER-NAME> ]
[ username <USER-NAME> ] [ ip-address <ADRR> ]
```

Parameters

<SERVER-NAME> – OpenVPN, PPTP or L2TP over IPsec server profile name;

<USER-NAME> – OpenVPN, PPTP or L2TP over IPsec user name;

<ADDR> – OpenVPN, PPTP or L2TP over IPsec user IP address.

When executing the command without a parameter, all the counters of OpenVPN, PPTP and L2TP over IPsec user connections will be reset.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear remote-access counters
```

clear remote-access session

This command ends the connection for OpenVPN, PPTP and L2TP over IPsec users.

Syntax

```
clear remote-access session [ pptp | l2tp | openvpn] [ server <SERVER-NAME> ] [ username
<USER-NAME> ] [ip-address <ADRR> ]
```

Parameters

<SERVER-NAME> – OpenVPN, PPTP or L2TP over IPsec server profile name;

<USER-NAME> – OpenVPN, PPTP or L2TP over IPsec user name;

<ADDR> – OpenVPN, PPTP or L2TP over IPsec user IP address. When executing the command without a parameter, all OpenVPN, PPTP and L2TP over IPsec connections will be ended.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear remote-access session
```

description

This command changes the description of the OpenVPN, PPTP and L2TP profile over IPsec servers.

The use of a negative form (no) of the command removes a profile description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – profile description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

CONFIG-OPENVPN-SERVER

Example

Set the description for PPTP server profile:

```
esr(config-pptp-server)# description "Our remote workers"
```

enable

This command activates the configured remote access server profile.

The use of a negative form (no) of the command disables the configurable profile.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

CONFIG-OPENVPN-SERVER

Example

```
esr(config-pptp-server)# enable
```

encryption mppe

This command enables Microsoft Point-to-Point Encryption (MPPE) encryption for PPTP connections.

The use of a negative form (no) of the command disables the encryption.

Syntax

```
[no] encryption mppe
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-PPTP-SERVER

Example

```
esr(config-pptp-server)# encryption mppe
```

remote-access

This command creates a remote access server profile.

The use of a negative form (no) of the command removes a specified profile.

Syntax

```
[no] remote-access <SERVER-TYPE> <NAME>
```

Parameters

<SERVER-TYPE> – remote access server type. May take following values: l2tp, openvpn, pptp

<NAME> – remote access server profile name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# remote-access l2tp remote-workers
esr(config-l2tp-server)#
```

show remote-access configuration

The command displays the parameters of the OpenVPN, PPTP and L2TP profiles over IPsec servers.

Syntax

```
show remote-access configuration { pptp | l2tp | openvpn } [ <NAME> ]
```

Parameters

<SERVER-NAME> – OpenVPN, PPTP or L2TP over IPsec server profile name;

When executing the command without a parameter, all OpenVPN, PPTP and L2TP over IPsec server parameters will be shown.

Required privilege level

10

Command mode

ROOT

Example

```
esr# show remote-access configuration pptp pptp1
State:                     Enabled
Description:               --
Security zone:             trusted
Authentication mode:       local
MTU:                       1500
Local address:             192.168.1.1
Remote address:            rem_pptp(10.0.10.20-10.0.10.40)
Outside address:           115.0.0.1
DNS server:                --
WINS server:               --
   Users
   ~~~~~
#     Name                   State      Encrypted password
---   --------------------   --------   ------------------------------
0     pptp                   Enabled    8CB5107EA7005AFF
1     petr                   Enabled    CCE5513EE45A1EAC
```

show remote-access counters

This command displays the connection counters for OpenVPN, PPTP and L2TP over IPsec users.

Syntax

```
show remote-access counters [ pptp | l2tp | openvpn ] [ server <SERVER-NAME> ] [ username
<USER-NAME> ] [ ip-address <ADRR> ]
```

Parameters

<SERVER-NAME> – PPTP or L2TP over IPsec server profile name;

<USER-NAME> – OpenVPN, PPTP or L2TP over IPsec user name;

<ADDR> – OpenVPN, PPTP or L2TP over IPsec user IP address.

When executing the command without a parameter, all the counters of OpenVPN, PPTP and L2TP over IPsec user connections will be shown.

Required privilege level

10

Command mode

ROOT

Example

```
esr# show remote-access counters
User            IP-address        UC recv      Bytes recv   Err recv     MC recv
-------------   ---------------   ----------   ----------   ----------   ----------
ivan            10.20.20.5        262          25365        0            0
fedor           20.20.20.160      59           5236         0            0
User            IP-address        UC sent      Bytes sent   Err sent
-------------   ---------------   ----------   ----------   ----------
ivan            10.20.20.5        249          29298        0
fedor           20.20.20.160      16           739          0
esr# show remote-access counters l2tp
PPTP Server: remote-workers
User: ivan(10.20.20.5)
 Packets received:               231
 Bytes received:                 22229
 Dropped on receive:             0
 Receive errors:                 0
 Multicasts received:            0
 Receive length errors:          0
 Receive buffer overflow errors: 0
 Receive CRC errors:             0
 Receive frame errors:           0
 Receive FIFO errors:            0
 Receive missed errors:          0
 Receive compressed:             0
 Packets transmitted:            189
 Bytes transmitted:              21858
 Dropped on transmit:            0
 Transmit errors:                0
 Transmit aborted errors:        0
 Transmit carrier errors:        0
 Transmit FIFO errors:           0
 Transmit heartbeat errors:      0
 Transmit window errors:         0
 Transmit comressed:             0
 Collisions:                     0
```

show remote-access status

This command displays the OpenVPN, PPTP and L2TP over IPsec user connections status.

Syntax

```
show remote-access status [ pptp | l2tp | openvpn ] [ server <SERVER-NAME> ] [ username
<USER-NAME> ] [ ip-address <ADRR> ]
```

Parameters

<SERVER-NAME> – OpenVPN, PPTP or L2TP over IPsec server profile name;

<USER-NAME> – OpenVPN, PPTP or L2TP over IPsec user name;

<ADDR> – OpenVPN, PPTP or L2TP over IPsec user IP address.

When executing the command without a parameter, all the statuses of OpenVPN, PPTP and L2TP over IPsec user connections will be shown.

Required privilege level

10

Command mode

ROOT

Example

```
esr# show remote-access status
User                IP-address        Server
----------------    ---------------   --------------------------------------
ivan                10.20.20.5        pptp(remote-workers)
fedor               20.20.20.160      l2tp(remote-workers-l2tp)
Count sessions: 2
```

L2TP over IPsec/PPTP server configuration

authentication mode

This command sets the authentication mode for remote users connecting via PPTP or L2TP over IPsec.

The use of a negative form (no) of the command removes a set mode.

Syntax

```
authentication mode { local | radius }
```

```
no authentication mode
```

Parameters

- local - authentication mode using the local user base of the configured profile.
- radius - the mode in which user authentication passes through a RADIUS server.

Required privilege level

15

Command mode

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Example

```
esr(config-pptp-server)# authentication mode local
```

authentication method

This command allows the usage of the authentication mode for remote users connecting via PPTP or L2TP over IPsec.

The use of a negative form (no) of the command removes a set mode.

Syntax

```
[no] authentication method <METHOD>
```

Parameters

<METHOD> – authentication method, possible values: [chap, mschap, mschap-v2, eap, pap].

Default value

Only chap is allowed

Required privilege level

10

Command mode

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Example

```
esr(config-pptp-server)# authentication method mschap
```

dns-servers

This command specifies the list of DNS servers that remote users using PPTP and L2TP over IPsec will use.

The use of a negative form (no) of the command removes configured DNS server addresses.

Syntax

```
dns-servers object-group <NAME>
no dns-servers
```

Parameters

<NAME> – name of IP addresses profile that contains addresses of required DNS servers, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Example

```
esr(config-pptp-server)# dns-servers object-group pptp_dns
```

dscp

The command sets the DSCP code value for the use in IP headers of PPTP and L2TP over IPsec server outgoing packets.

The use of a negative form (no) of the command sets the default DSCP value.

Syntax

```
dscp <DSCP>
no dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Default value

32

Required privilege level

10

Command mode

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Example

```
esr(config-pptp-server)# dscp 40
```

ipsec authentication method

This command selects the key authentication method for the IKE connection. Message authentication by key is used when an IKE connection is established. The key is set by the 'ipsec authentication pre-shared-key' command (see subsection ipsec authentication pre-shared-key).

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipsec authentication method pre-shared-key
```

```
no ipsec authentication method
```

Parameters

pre-shared-key – authentication method using pre-received encryption keys.

Required privilege level

15

Command mode

CONFIG-L2TP-SERVER

Example

```
esr(config-l2tp-server)# ipsec authentication method pre-shared-key
```

ipsec authentication pre-shared-key

This command specifies a shared secret authentication key that should be the same for both parties of the tunnel.

The use of a negative form (no) of the command removes a set key.

Syntax

```
ipsec authentication pre-shared-key { ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }|
hexadecimal {<HEX> | encrypted <ENCRYPTED-HEX> } }
```

```
no ipsec authentication pre-shared-key
```

Parameters

<TEXT> – string [1..64] ASCII characters.

<HEX> – number, [1..32] bytes size, set by the string of [2..128] characters in hexadecimal format (0xYYYY ...) or (YYYY ...).

<ENCRYPTED-TEXT> – encrypted password, [1..32] bytes size, set by the string of [2..128] characters.

<ENCRYPTED-TEXT> – encrypted number, [2..64] bytes size, set by the string of [2..256] characters.

Default value

None

Required privilege level

15

Command mode

CONFIG-L2TP-SERVER

Example

```
esr(config-l2tp-server)# ipsec authentication pre-shared-key ascii-text password
```

local-address

This command specifies the IP address used by the PPTP or L2TP over IPsec server as the local IP address of the tunnel.

The use of a negative form (no) of the command removes configured tunnel local IP address.

Syntax

```
local-address { object-group <NAME> | ip-address <ADDR> }
```

```
no local-address
```

Parameters

<NAME> – name of IP addresses profile that contains local IP address of the tunnel, set by the string of up to 31 characters.

<ADDR> – local IP address of the tunnel, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Example

```
esr(config-pptp-server)# local-address object-group pptp_local
```

mtu

This command specifies the MTU for the interfaces that will be created when remote users connect using PPTP and L2TP over IPsec.

The use of a negative form (no) of the command sets the default MTU value.

Syntax

`mtu <MTU>`

`no mtu`

Parameters

<MTU> – MTU value, takes values in the range of [1280..1500].

Default value

1500

Required privilege level

10

Command mode

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Example

```
esr(config-pptp-server)# mtu 1400
```

outside-address

This command specifies the IP address that the PPTP or L2TP over IPsec server will listen on for incoming connections.

The use of a negative form (no) of the command removes the configured listening address.

Syntax

`outside-address { object-group <NAME> | ip-address <ADDR> | interface { <IF> | <TUN> } }`

`no outside-address`

Parameters

<NAME> – the name of the IP address profile containing the address that PPTP or L2TP over IPsec will listen on for incoming connections, is specified in a string of up to 31 characters.

<ADDR> – the IP address that PPTP or L2TP over IPsec will listen on for incoming connections, defined as AAA.BBB.CCC.DDD, where each part takes the values [0..255].

<IF> – an interface, specified in the form described in Section Types and naming order of router interfaces.

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

Required privilege level

10

Command mode

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Example

```
esr(config-pptp-server)# outside-address object-group pptp_outside
```

remote-address

This command specifies a list of IP addresses from which PPTP or L2TP over IPsec server issues dynamic IP addresses to remote users.

The use of a negative form (no) of the command removes the remote user IP addresses list.

Syntax

```
remote-address { object-group <NAME>| address-range <FROM-ADDR>-<TO-ADDR> }
no remote-address
```

Parameters

<NAME> – name of IP addresses profile that contains remote user IP addresses list, set by the string of up to 31 characters.

<FROM-ADDR> – range starting IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<TO-ADDR> – range ending IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Example

```
esr(config-pptp-server)# remote-address object-group pptp_remote
```

remote network

The command is used to set an IP address of a subnet available when a dynamic PPTP/L2TP tunnel is created.

The use of a negative form (no) of the command removes an IP address of a subnet available via dynamic PPTP/L2TP tunnel creation.

Syntax

```
remote network <ADDR/LEN>
no remote network
```

Parameters

<ADDR/LEN> – IP subnet of a recipient. The parameter is defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32].

Required privilege level

10

Command mode

CONFIG-PPP-USER

Example

```
esr(config-ppp-user)# remote network 192.168.54.0/24
```

remote networks

The command is used to set a list of IP addresses of subnets available when a dynamic PPTP/L2TP tunnel is created.

The use of a negative form (no) of the command removes an IP address of a subnet available via dynamic PPTP/L2TP tunnel creation.

Syntax

```
remote networks <OBJ-GROUP-NETWORK-NAME>
no remote network
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – IP/IPv6 addresses profile name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-PPP-USER

Example

```
esr(config-ppp-user)# remote network 192.168.54.0/24
```

username

This command creates a user to connect to PPTP or L2TP over IPsec servers. After executing command, the router enters the PPP user password configuration mode.

The use of a negative form (no) of the command removes a specified user.

The command sets the command line mode to PPTP USER or L2TP USER depending on the current command mode.

Syntax

```
[no] username <NAME>
```

Parameters

<NAME> – user name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Example

```
esr(config-pptp-server)# username fedor
esr(config-pptp-user)#
```

wins-servers

This command specifies the list of WINS servers that remote users using PPTP and L2TP over IPsec will use.

The use of a negative form (no) of the command removes configured WINS server addresses.

Syntax

`wins-servers object-group <NAME>`

`no wins-servers`

Parameters

<NAME> – name of IP addresses profile that contains addresses of required WINS servers, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Example

```
esr(config-pptp-server)# wins-servers object-group l2tp_wins
```

OpenVPN server configuration

address-range

This command specifies the IP addresses list from which dynamic IP addresses are leased to remote users in L2 mode by OpenVPN server.

The use of a negative form (no) of the command removes the remote user IP addresses list.

Syntax

`address-range <FROM-ADDR>-<TO-ADDR>`

`no address-range`

Parameters

<FROM-ADDR> – range starting IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<TO-ADDR> – range ending IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# address-range 192.168.1.10-192.168.1.250
```

authentication algorithm

This command defines OpenVPN clients authentication algorithm.

The use of a negative form (no) of the command sets the default authentication mode.

Syntax

```
authentication algorithm <ALGORITHM>
no authentication-algorithm
```

Parameters

<ALGORITHM> – authentication algorithm:

- 8-128 bits key size: md4, rsa-md4, md5, rsa-md5, mdc2, rsa-mdc2
- 8-160 bits key size: sha, sha1, rsa-sha, rsa-sha1, rsa-sha1-2, dsa, dsa-sha, dsa-sha1, dsa-sha1-old, ripemd160, rsa-ripemd160, ecdsa-with-sha1
- 8-224 bits key size: sha-224, rsa-sha-224
- 8-256 bits key size: sha-256, rsa-sha-256
- 8-384 bits key size: sha-384, rsa-sha-384
- 8-512 bits key size: sha-512, rsa-sha-512, whirlpool

Default value

sha

Required privilege level

15

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# authentication algorithm cleartext
```

bridge-group

This command includes client connections via OpenVPN to the L2 domain.

The use of a negative form (no) of the command excludes connections from the L2 domain.

Syntax

```
bridge-group <BRIDGE-ID>
no bridge-group
```

Parameters

<BRIDGE-ID> – bridge identifying number. Specified in the form described in Section Types and naming order of router interfaces.

Required privilege level

10

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# bridge-group 15
```

certificate

This command specifies certificates and keys. Certificates and keys must be previously copied to the router using the *copy* command described in section copy.

The use of a negative form (no) of the command removes a specified certificate from the profile.

Syntax

```
certificate <CERTIFICATE-TYPE> <NAME>
no certificate <CERTIFICATE-TYPE>
```

Parameters

<CERTIFICATE-TYPE> – certificate or key type, may take the following values:

- ca – Certificate Authority;
- crl – Certificate Revocation List;
- dh – Diffie-Hellman key;
- server-crt – public server certificate;
- server-key – private server key;
- ta – HMAC key
- client-key – OPENVPN client private key;
- client-crt – OPENVPN client certificate.

<NAME> – certificate or key name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# certificate ca ca.crt
```

client-isolation

This commands enables blocking of data transfer between clients.

The use of a negative form (no) of the command removes blocking.

Syntax

```
[no] client-isolation
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# client-isolation
```

client-max

This command sets the maximum number of simultaneous user sessions.

The use of a negative form (no) of the command sets the default value.

Syntax

```
client-max <VALUE>
no client-max
```

Parameters

<VALUE> – maximum amount of users, takes values of [1..65535].

Default value

Not limited.

Required privilege level

10

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# clients-max 500
```

compression

This command enables the mechanism of transmitted data compression between clients and the OPENVPN server.

The use of a negative form (no) of the command disables the mechanism of transmitted data compression.

Syntax

```
[no] compression
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# compression
```

dns-server

This command specifies the list of DNS servers that will be used by remote users.

The use of a negative form (no) of the command removes configured DNS server addresses.

Syntax

```
dns-server <ADDR>
no dns-server { <ADDR> | all }
```

Parameters

<ADDR> – DNS server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

all – remove all configured IP address ranges.

Required privilege level

10

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# dns-server 1.1.1.1
```

duplicate-cn

The command allows connecting several users with one certificate.

The use of a negative form (no) of the command prohibits the use of the same certificate by several users.

Syntax

`[no] duplicate-cn`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# duplicate-cn
```

encryption algorithm

This command selects the encryption algorithm used when data transmission.

The use of a negative form (no) of the command disables the encryption.

Syntax

`encryption algorithm <ALGORITHM>`

`no encryption algorithm`

Parameters

<ALGORITHM> – encryption protocol identifier, may take following values: des, blowfish128, aes128, des-ede, aes192, 3des, desx, aes256.

Default value

Encryption disabled.

Required privilege level

15

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# encryption algorithm aes128
```

ip address

The command sets a static IP address for a specified user. The use of a negative form (no) of the command removes a client's static IP address.

Syntax

```
[no] ip address <ADDR>
```

Parameters

<ADDR> – IP address set in the following format:

AAA.BBB.CCC.DDD – IP address of a subnet with a mask in the prefix form where AAA-DDD take values [0..255].

Required privilege level

15

Command mode

CONFIG-OPENVPN-USER

Example

```
esr(config-openvpn-server)# username client
esr(config-openvpn-user)# ip address 10.10.100.15
```

login authentication

This command activates the user authentication list to authorize users.

The default configuration includes a list named 'default'; the list contains one authentication method – 'local'.

The use of a negative form of the command (no) disables authentication list.

Syntax

```
login authentication <NAME>
no login authentication
```

Parameters

<NAME> – list name, set by the string of up to 31 characters.

Default value

default

Required privilege level

15

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# login authentication OPENVPN-LIST
```

network

This command defines the subnet from which IP addresses are leased to users. The first IP address on the subnet is the gateway for user sessions.

The use of a negative form (no) of the command removes this subnet.

Syntax

`network <ADDR/LEN>`

`no network`

Parameters

<ADDR/LEN> – subnet IPaddress, set in one of the following formats:

- BBB.CCC.DDD/NN – network IP address with prefix mask, where AAA-DDD take values of [0..255] and EE takes values of [1..32].

Required privilege level

10

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# network 192.168.25.0/24
```

port

This command sets the TCP/UDP port that the OpenVPN server will listen on.

The use of a negative form (no) of the command sets the default value.

Syntax

```
port <PORT>
no port
```

Parameters

<PORT> – TCP/UDP port, takes values of [1..65535].

Default value

1194

Required privilege level

15

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# port 5000
```

protocol

The command sets encapsulation protocol.

The use of a negative form (no) of the command sets the default value.

Syntax

```
protocol <PROTOCOL>
no protocol
```

Parameters

<TYPE> – encapsulation type, possible values:

- TCP – encapsulation in TCP segments;
- UDP – encapsulation in UDP datagrams.

Default value

Stopped.

Required privilege level

15

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# protocol udp
```

redirect-gateway

This command enables the default route advertising for OpenVPN connections, which leads to the replacement of the default route on the client side. The new default gateway will be the OpenVPN server IP address.

The use of a negative form (no) of the command disables the default route advertising.

Syntax

```
[no] redirect-gateway
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# redirect-gateway
```

route

This command enables advertising of the specified subnets, the IP address of the OpenVPN server is the gateway (the first IP address from the subnet specified using the network command described in the section network).

The use of a negative form (no) of the command disables specified subnets advertising.

Syntax

```
route <ADDR/LEN>
```

```
no route { <ADDR/LEN> | all }
```

Parameters

<ADDR/LEN> – subnet IP address set in the following format:

AAA.BBB.CCC.DDD/EE – network IP address with prefix mask, where AAA-DDD take values of [0..255] and EE takes values of [1..32].

Required privilege level

10

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# route 192.168.25.0/24, 192.168.26.0/24
```

timers holdtime

This command sets time interval after which the opposing party is considered to be unavailable. The timer starts after establishing a neighborhood relationship and starts counting from 0. The timer is reset when each reply to a keepalive message from the opposite side is received. It is recommended to set the timer value to _3 * keepalive_.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timers holdtime <TIME>
```

```
no timers holdtime
```

Parameters

<TIME> – time in seconds, takes values of [1..65535].

Default value

120

Required privilege level

10

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# timers holdtime 360
```

timers keepalive

This command sets the time interval after which the connection with the opposing party will be checked.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timers keepalive <TIME>
no timers keeaplive
```

Parameters

<TIME> – time in seconds, takes values of [1..65535].

Default value

10

Required privilege level

10

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# timers keepalive 120
```

subnet

This command defines the subnet for the specified user of the OpenVPN server

The use of a negative form (no) of the command removes a bind to a specified subnet.

Syntax

```
[no] subnet <ADDRLEN>
```

Parameters

<ADDR/LEN> – subnet IP address set in the following format:

AAA.BBB.CCC.DDD/NN – network IP address with prefix mask, where AAA-DDD take values of [0..255] and EE takes values of [1..32].

Required privilege level

10

Command mode

CONFIG-OPENVPN-USER

Example

```
esr(config-openvpn-server)# username client
esr(config-openvpn-user)# subnet 192.168.25.128/28
```

tunnel

This command defines type of connection with a private network via OpenVPN server.

The use of a negative form (no) of the command removes a current value.

Syntax

```
tunnel <TYPE>
```

```
no tunnel
```

Parameters

<TYPE> – encapsulation protocol, takes the following values:

- ip – point-to-point connection;
- ethernet – L2 domain connection.

Default value

None

Required privilege level

10

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# tunnel ip
```

username

The command allows switching to a specified OpenVPN user's configuration mode.

The use of a negative form (no) of the command returns default user settings.

Syntax

`[no] username { <NAME> | all }`

Parameters

<NAME> – user name, set by the string of up to 31 characters.

all — the key used to delete all users created before.

Required privilege level

15

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# username client
esr(config-openvpn-user)#
```

wins-server

This command specifies the list of WINS servers that will be used by remote users.

The use of a negative form (no) of the command removes configured WINS server addresses.

Syntax

`wins-server <ADDR>`

`no wins-server { <ADDR> | all }`

Parameters

<ADDR> – WINS server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

all – remove all configured DNS server IP addresses.

Required privilege level

10

Command mode

CONFIG-OPENVPN-SERVER

Example

```
esr(config-openvpn-server)# wins-servers 1.1.1.1
```

23 Routing

- General routing settings
- General commands for routes advertising and reception
- Policy-based routing (PBR)
- Key bindings configuration
- Event tracking object configuration
- BFD configuration
- IPv4/IPv6 static routes configuration
- BGP configuration
- RIP and RIPNG configuration
- OSPF and OSPFv3 configuration
- IS-IS protocol configuration

General routing settings

- authentication algorithm
- authentication key
- enable
- graceful-restart
- graceful-restart timeout
- ip path-mtu-discovery
- ip protocols max-routes
- ip protocols preference
- ipv6 protocols max-routes
- ipv6 tcp adjust-mss
- show ip protocols
- show ip route
- show ipv6 protocols
- show ipv6 route

authentication algorithm

This command defines authentication algorithm.

The use of a negative form (no) of the command disables the authentication.

Syntax

```
authentication algorithm <ALGORITHM>
```

```
no authentication algorithm
```

Parameters

<ALGORITHM> – authentication algorithm:

- cleartext – password, transmitted in unencrypted form (available only for CONFIG-RIP and CONFIG_OSPF-VLINK modes);
- md5 – password is hashed by md5 algorithm.

Required privilege level

10

Command mode

CONFIG-RIP

CONFIG-BGP

CONFIG-BGP-VRF

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-OSPF-VLINK

Example

```
esr(config-rip)# authentication algorithm cleartext
```

authentication key

This command sets a password for authentication with a neighbor.

The use of a negative form (no) of the command removes the password.

Syntax

`authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }`

`no authentication key`

Parameters

<CLEAR-TEXT> – password, set by the string of 8 to 16 characters.

<ENCRYPTED-TEXT> – encrypted password of [8..16] bytes ([16..32] characters) in hexadecimal format (0xYYYY...) or (YYYY...).

Required privilege level

10

Command mode

CONFIG-RIP

CONFIG-BGP

CONFIG-BGP-VRF

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-OSPF-VLINK

Example

```
esr(config-bgp-af)# authentication key ascii-text 123456789
esr(config-bgp-af)# authentication key ascii-text encrypted CDE65039E5591FA3F1
```

enable

This command enables the configurable routing protocol, area, virtual connection, neighborhood.

The use of a negative form (no) of the command disables RIP.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-RIP

CONFIG-RIPNG

CONFIG-OSPF

CONFIG-OSPF-AREA

CONFIG-OSPF-VLINK

CONFIG-OSPFV3

CONFIG-OSPFV3-AREA

CONFIG-OSPFV3-VLINK

CONFIG-ISIS

CONFIG-BGP

CONFIG-BGP-VRF

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

Example 1

```
esr(config-rip)# enable
```

Example 2

```
esr(config-ospf)# enable
```

Example 3

```
 esr(config-isis)# enable
```

Example 4

```
esr(config-bgp-neighbor)# enable
```

graceful-restart

This command enables the mechanism to maintain an active neighborhood state with route information preserved at the time the routing protocol is restarted.

The use of a negative form (no) of the command disables this mechanism.

Syntax

```
[no] graceful-restart
```

Parameters

The command does not contain parameters

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-OSPF

CONFIG-OSPFV3

Example

```
esr(config-bgp-neighbor)# graceful-restart
```

graceful-restart timeout

This command sets the time interval during which to maintain the active state of the neighborhood at the time of the routing protocol restart. When the time interval expires, the opposite side is considered unavailable. Used in conjunction with the graceful-restart command.

The use of a negative form (no) of the command sets the default value.

Syntax

```
graceful-restart timeout <TIME>
```

```
no graceful-restart timeout
```

Parameters

<TIME> – time in seconds, takes values of [1..65535];

Default value

120

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-OSPF

CONFIG-OSPFV3

Example

```
esr(config-bgp-neighbor)# graceful-restart timeout 60
```

ip path-mtu-discovery

This command enables/disables PMTU search for TCP, SCTP, DCCP.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip path-mtu-discovery <ACT>
```

```
no ip path-mtu-discovery
```

Parameters

<ACT> – allocated action:

- enable – enables PMTU search for TCP, SCTP, DCCP;
- disable – disables PMTU search for TCP, SCTP, DCCP.

Default value

PMTU search enabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip path-mtu-discovery disable
```

ip protocols max-routes

This command configures the routing tables capacity.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip protocols <PROTOCOL> max-routes <VALUE>
```

```
no ip protocols <PROTOCOL> max-routes
```

Parameters

<PROTOCOL> – protocol type, may take following values: rip (only in global mode), ospf, isis, bgp;

<VALUE> – amount of routes in the routing table, takes values in the range of:

- BGP
  - ESR-1000/1200/1500/1511/1700/3100 [1..5000000];
  - ESR-20/21/100/200 [1..2500000],
  - ESR-10/12V/12VF/14VF [1.. 1000000].
- OSPF and IS-IS
  - ESR-1000/1200/1500/1511/1700/3100 [1..500000];
  - ESR-20/21/100/200 [1..300000];
  - ESR-10/12V/12VF/14VF [1..30000].
- RIP
  - ESR-20/21/100/200/1000/1200/1500/1511/1700/3100 [1..10000];
  - ESR-10/12V/12VF/14VF [1..1000].

Default value for the global mode

BGP

- ESR-1000/1200/1500/1511/1700/3100 (5000000);
- ESR-20/21/100/200 (2500000);
- ESR-10/12V/12VF/14VF (1000000).

OSPF and IS-IS

- ESR-1000/1200/1500/1511/1700/3100 (500000);
- ESR-20/21/100/200 (300000);
- ESR-10/12V/12VF/14VF (30000).

RIP

- ESR-20/21/100/200/1000/1200/1500/1511/1700/3100 (10000);
- ESR-10/12V/12VF/14VF (1000).

Required privilege level

10

Command mode

CONFIG

CONFIG-VRF

Example

```
esr(config)# ip protocols ospf max-routes 4400
```

ip protocols preference

This command configures the priority of routing protocols for the main routing table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip protocols <PROTOCOL> preference <VALUE>
```

```
no ip protocols <PROTOCOL> preference
```

Parameters

<PROTOCOL> – protocol type, may take values: static, rip, ospf, isis, bgp, dhcp, l2tp, pppoe, pptp;

<VALUE> – protocol precedence, takes values in the range of [1..255].

Default value

BGP (170)

IS-IS (160)

OSPF (150)

RIP (100)

DHCP (40)

L2TP (50)

PPPoE (50),

PPTP (50)

Static (1)

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip protocols ospf preference 44
```

ipv6 protocols max-routes

This command configures the IPv6 routing tables capacity.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 protocols <PROTOCOL> max-routes <VALUE>
no ipv6 protocols <PROTOCOL> max-routes
```

Parameters

<PROTOCOL> – protocol type, may take following values: rip (only in global mode), ospf, isis, bgp;

<VALUE> – amount of routes in the routing table, takes values in the range of:

- BGP
  - ESR-1000/1200/1500/1511/1700/3100 [1..5000000];
  - ESR-20/21/100/200 [1..2500000],
  - ESR-10/12V/12VF/14VF [1.. 1000000].
- OSPFv3 and IS-IS
  - ESR-1000/1200/1500/1511/1700/3100 [1..500000];
  - ESR-20/21/100/200 [1..300000];
  - ESR-10/12V/12VF/14VF [1..30000].
- IPv6 BGP
  - ESR-1000/1200/1500/1511/1700/3100 [1..5000000];
  - ESR-1000/1200/1500 [1..3000000];
  - ESR-10/12V/12VF/14VF/20/21/100/200 [1..1500000].

Default value for the global mode

OSPFv3

- ESR-1000/1200/1500/1511/1700/3100 (500000);
- ESR-20/21/100/200 (300000);
- ESR-10/12V/12VF/14VF [1..30000].

IPv6 BGP

- ESR-1700/3100 (5000000),
- ESR-1000/1200/1500/1511/1700/3100 (5000000),
- ESR-20/21/100/200 [1..2500000],
- ESR-10/12V/12VF/14VF [1.. 100000];

Required privilege level

10

Command mode

CONFIG

CONFIG-VRF

Example

```
esr(config)# ipv6 protocols ospf max-routes 4400
```

ipv6 tcp adjust-mss

This command overrides the value of the MSS (Maximum segment size) field in incoming TCP packets.

The use of a negative form (no) of the command disables MSS field value correction.

Syntax

```
ipv6 tcp adjust-mss <MSS>
no ipv6 tcp adjust-mss
```

Parameters

<MSS> – MSS value, takes values in the range of [40..1940].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

Example

```
esr(config-if-gi)# ipv6 tcp adjust-mss 1400
```

show ip protocols

This command displays information about IP routing protocol settings.

Syntax

```
show ip protocols [ <PROTOCOL> ] [ vrf <VRF> ]
```

Parameters

<PROTOCOL> – the routing protocol by which information should be displayed:

- bgp;
- ospf;
- isis;
- rip;
- static;
- dhcp;
- pptp;
- pppoe;
- l2tp.

Without specifying the routing protocol, information about all routing protocols configuration is displayed.

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# sh ip protocols
BGP:
    Max routes:    --
    Preference:    170
OSPF:
    Max routes:    --
    Preference:    150
RIP:
    Max routes:    --
    Preference:    100
Static:
    Preference:    1
```

show ip route

This command displays device routing table. If the <SUBNET> parameter is specified, the routes to this subnet are displayed in detail. If the <VRF> parameter is specified, the command displays the routing table of the specified VRF instance.

Syntax

```
show ip route [ vrf <VRF> ] [ { <SUBNET> [ long-prefix ] | all | summary | arp-proxy |
<PROTOCOL> } ]
```

Parameters

<VRF> – VRF name, set by the string of up to 31 characters.

<SUBNET> – destination address, optional parameter, can be specified in the following formats:

- BBB.CCC.DDD – host IP address, where each part takes values of [0..255].
- BBB.CCC.DDD/NN – network IP address with prefix mask, where AAA-DDD take values of [0..255] and NN takes values of [1..32];
- all – displays information about all routes, including those not selected for FIB;
- long-prefix – displays information about routes to networks that are a given subnet;
- summary – displays summary statistics of routing protocols;
- arp-proxy – displays information about configured arp-nat-proxy pools on interfaces and tunnels;
- <PROTOCOL> – filter by protocol type (bgp, connected, ospf , isis, rip, static).

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip route
Codes: C - connected, S - static, R - RIP derived,
        O - OSPF derived, IA - OSPF inter area route,
        E1 - OSPF external type 1 route, E2 - OSPF external type 2 route
        B - BGP derived, D - DHCP derived, K - kernel route,
        * - FIB route
C     * 192.168.1.0/24    [0/0]   dev br1                          [direct 01:14:16]
C     * 10.100.100.0/24   [0/0]   dev gi1/0/5                      [direct 01:14:17]
esr# show ip route summary
Direct Connected: 12
Static:           46
RIP:              0
OSPF:             2000
BGP:              100000
```

show ipv6 protocols

This command displays information about IPv6 routing protocol settings.

Syntax

```
show ipv6 protocols [ <PROTOCOL> ] [ vrf <VRF> ]
```

Parameters

<PROTOCOL> – the routing protocol by which information should be displayed:

- bgp;
- ospf;
- isis;
- static;
- dhcp;
- pppoe;
- pptp;
- l2tp.

Without specifying the routing protocol, information about all routing protocols configuration is displayed.

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# sh ipv6 protocols
BGP:
    Max routes:    --
    Preference:    170
OSPF:
    Max routes:    --
    Preference:    150
Static:
    Preference:    1
```

show ipv6 route

The command is intended for viewing device routing table. If the <SUBNET> parameter is specified, the routes to this subnet are displayed in detail. If the <VRF> parameter is specified, the command displays the routing table of the specified VRF instance.

Syntax

```
show ipv6 route [ vrf <VRF> ] [ { <SUBNET> | all | summary | <PROTOCOL> } ]
```

Parameters

<VRF> – VRF name, set by the string of up to 31 characters.

<SUBNET> – destination address, optional parameter, can be specified in the following formats:

- X:X:X:X::X – host IPv6 address, where each X part takes values in hexadecimal format [0..FFFF].
- <IPV6-ADDR/LEN> – IP address and mask of a subnet, defined as X:X:X:X::X/EE where each X part takes values in hexadecimal format [0..FFFF] and EE takes values of [1..128];
- all – displays information about all routes, including inactive ones;
- summary – displays summary statistics of routing protocols;
- <PROTOCOL> – filter by protocol type (bgp, connected, ospf , isis, static).

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 route
 Codes: C - connected, S - static, R - RIP derived,
        O - OSPF derived, IA - OSPF inter area route,
        E1 - OSPF external type 1 route, E2 - OSPF external type 2 route
        B - BGP derived, D - DHCP derived, K - kernel route,
        * - FIB route
S     * ::/0              [1/0]   via fc00::1 on gi1/0/5            [static 03:16:23]
S     * 2001::/120        [1/6]   dev gi1/0/5                       [static 03:16:23]
C     * fc00::/120        [0/0]   dev gi1/0/5                       [direct 03:16:23]
S     * fc00:3::1/128     [1/0]   via fc00::1 on gi1/0/5            [static 03:16:23]
esr# show ipv6 route summary
Direct Connected: 1
Static:           3
RIP:              0
OSPF:             0
BGP:              0
```

General commands for routes advertising and reception

- default-information-originate
- description
- ip prefix-list
- ipv6 prefix-list
- network
- permit/deny
- prefix-list
- redistribute bgp
- redistribute connected
- redistribute ipv6 bgp
- redistribute ipv6 ospf
- redistribute isis
- redistribute ospf
- redistribute rip
- redistribute static

default-information-originate

This command enables the generation and sending of the default route, if it is registered in the routing table of the FIB, for:

- NSSA areas (route advertising) as NSSA-LSA;
- BGR IPv4/IPv6 address family.

The use of a negative form (no) of the command disables the route generation by default.

Syntax

```
[no] default-information-originate
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-BGP

CONFIG-BGP-VRF

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

Example

```
esr(config-ospf-area)# default-information-originate
```

description

This command changes the description of a configurable list of IP subnets or IPv6 subnets.

The use of a negative form (no) of the command removes a specified description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – IP or IPv6 subnets list description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-PL

CONFIG-IPV6-PL

Example

```
esr(config-route-map-rule)# description "Drop Local NETs"
```

ip prefix-list

This command creates IP subnets list that will be used for further filtration of advertised and received IP routes.

The use of a negative form (no) of the command removes prefixes list.

Syntax

```
[no] ip prefix-list <NAME>
```

Parameters

<NAME> – name of a subnet list being configured, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip prefix-list ospf_in
```

ipv6 prefix-list

This command creates IPv6 subnets list that will be used for further filtration of advertised and received IP routes.

The use of a negative form (no) of the command removes prefixes list.

Syntax

```
[no] ipv6 prefix-list <NAME>
```

Parameters

<NAME> – name of a subnet list being configured, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 prefix-list ospfv3_in
```

network

This command enables specified subnet advertising.

The use of a negative form (no) of the command disables specified subnet advertising.

Syntax

```
[no] network <ADDR/LEN>
```

Parameters

<ADDR/LEN> – IP subnet, set in the following format:

- BBB.CCC.DDD/EE – network IP address with prefix mask, where AAA-DDD take values of [0..255] and EE takes values of [1..32].
- X:X:X:X::X/EE – IPv6 address and mask of a subnet, where each X part takes values in hexadecimal format [0..FFFF] and EE takes values of [1..128].

Required privilege level

10

Command mode

CONFIG-RIP

CONFIG-RIPNG

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

Example

```
esr(config-bgp)# network 192.168.25.0/24
```

permit/deny

This command allows (permit) or forbids (deny) prefix lists.

Syntax

```
permit [ { object-group <OBJ-GROUP-NETWORK-NAME> | <ADDR/LEN> | <IPV6-ADDR/LEN> } ]
[ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ] } ]
```

```
deny [ { object-group <OBJ-GROUP-NETWORK-NAME> | <ADDR/LEN > | <IPV6-ADDR/LEN> } ] [ { eq
<LEN> | le <LEN> | ge <LEN> [ le <LEN> ] } ]
```

```
no { object-group <OBJ-GROUP-NETWORK-NAME> | prefix { <ADDR/LEN> | <IPV6-ADDR/LEN> } }
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – IP/IPv6 addresses profile name, set by the string of up to 31 characters;

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<LEN> – prefix length, takes values of [1..32] in prefix IP lists and [1..128] in prefix IPv6 lists;

eq – when specifying the command, the prefix length mast match the specified one;

le – when specifying the command, the prefix length mast be less than or match the specified one;

ge – when specifying the command, the prefix length mast be more than or match the specified one.

Required privilege level

10

Command mode

CONFIG-PL

CONFIG-IPV6-PL

Example

```
esr(config-pl)# permit static ge 24 le 28
```

prefix-list

This command adds subnet filtering for inbound or outbound updates.

The use of a negative form (no) of the command disables the filtration.

Syntax

```
prefix-list <PREFIX-LIST-NAME> { in | out }
no prefix-list { in| out }
```

Parameters

<PREFIX-LIST-NAME> – name of a subnet list being configured, set by the string of up to 31 characters:

in – incoming routes filtration;

out – advertised routes filtration.

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMALY

CONFIG-BGP-VRF-NEIGHBOR-FAMALY

CONFIG-OSPF

CONFIG-OSPFV3

CONFIG-RIP

CONFIG-RIPNG

Example

```
esr(config-rip)# prefix-list rip_in in
```

redistribute bgp

This command enables the advertising of BGP autonomous system routes.

The use of a negative form (no) of the command disables the advertising of BGP autonomous system routes.

Syntax

```
redistribute bgp <AS> [ route-map <NAME> ] [ <LEVEL> ]
```

```
no redistribute bgp <AS>
```

Parameters

<AS> – number of the autonomous system from which routes will be advertised, may take values [1..4294967295];

<NAME> – name of the route map that will be used for advertised BGP routes filtration and modification, set by the string of up to 31 characters.

<LEVEL> — IS-IS protocol level into which routes are advertised (only relevant in CONFIG-ISIS configuration mode):

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2;

Required privilege level

10

Command mode

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-OSPFV3

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

Example

```
esr(config-bgp-af)# redistribute bgp 30
esr(config-ospf)# redistribute bgp 35
esr(config-rip)# redistribute bgp 300
```

redistribute connected

This command enables advertising of directly connected subnets.

The use of a negative form (no) of the command disables advertising of directly connected subnets.

Syntax

```
redistribute connected [ route-map <NAME> ] [ <LEVEL> ]
no redistribute connected
```

Parameters

<NAME> – name of the route map that will be used for filtration and modification of advertised directly connected subnets, set by the string of up to 31 characters.

<LEVEL> — IS-IS protocol level into which routes are advertised (only relevant in CONFIG-ISIS configuration mode):

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2;

Required privilege level

10

Command mode

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-OSPFV3

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

Example

```
esr(config-rip)# redistribute connected
```

redistribute ipv6 bgp

This command enables the advertising of BGP autonomous system ipv6 routes.

The use of a negative form (no) of the command disables the advertising of BGP autonomous system ipv6 routes.

Syntax

```
redistribute ipv6 bgp <AS> [ route-map <NAME> ] [ <LEVEL> ]
no redistribute ipv6 bgp <AS>
```

Parameters

<AS> – number of the autonomous system from which ipv6 routes will be advertised, may take values [1..4294967295];

<NAME> – name of the route map that will be used for advertised BGP ipv6 routes filtration and modification, set by the string of up to 31 characters.

<LEVEL> — IS-IS protocol level into which routes are advertised (only relevant in CONFIG-ISIS configuration mode):

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2;

Required privilege level

10

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# redistribute ipv6 bgp 30 level 2
```

redistribute ipv6 ospf

This command enables advertising of routes from the OSPFv3 process database according to the selected conditions.

The use of a negative form (no) of the command disables the advertising of routes from the OSPFv3 process database.

Syntax

```
redistribute ipv6 ospf <ID> [ <ROUTE-TYPE1> ] [ <ROUTE-TYPE2> ] [ <ROUTE-TYPE3> ]
[ <ROUTE-TYPE4> ] [ route-map <NAME> ] [ <LEVEL> ]
```

```
no redistribute ipv6 ospf <ID>
```

Parameters

<ID> – process number, takes values of [1..65535].

<ROUTE-TYPE> – types of routes to be advertised:

- intra-area – OSPFv3 process routes advertising within a zone;
- inter-area – OSPFv3 process routes advertising between zones;
- external1 – OSPFv3 format 1 external routes advertising;
- external2 – OSPFv3 format 2 external routes advertising.

<NAME> – name of the route map that will be used for advertised OSPFv3 routes filtration and modification, set by the string of up to 31 characters.

<LEVEL> — IS-IS protocol layer into which routes are advertised:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2;

Required privilege level

10

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# redistribute ospf 10 external2
```

redistribute isis

This command enables advertising of routes from the IS-IS process database according to the selected conditions.
The use of a negative form (no) of the command disables advertising of routes from the ISIS route database.

Syntax

```
redistribute isis <ID> <ROUTE-TYPE> [ route-map <NAME> ]
```

```
no redistribute isis <ID>
```

Parameters

<ID> – process number, takes values of [1..65535].

<ROUTE-TYPE> – route type:

- level-1 – advertising the routes of the ISIS layer 1 process;
- level-2 – advertising the routes of the ISIS layer 2 process;
- inter-area – advertising the inter-area routes of the ISIS process.

<NAME> – name of the route map that will be used for advertised ISIS routes filtration and modification, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-OSPFV3

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

Example

```
esr(config-bgp-af)# redistribute isis 1 level-1 level-2
```

redistribute ospf

This command enables advertising of routes from the OSPF process database according to the selected conditions.

The use of a negative form (no) of the command disables the advertising of routes from the OSPF process database.

Syntax

```
redistribute ospf <ID> <ROUTE-TYPE1> [ <ROUTE-TYPE2> ] [ <ROUTE-TYPE3> ] [ <ROUTE-TYPE4> ] [ route-map <NAME> ] [ <LEVEL> ]
```

```
no redistribute ospf <ID>
```

Parameters

<ID> – process number, takes values of [1..65535].

<ROUTE-TYPE> – types of routes to be advertised:

- intra-area – OSPF process routes advertising within a zone;
- inter-area – OSPF process routes advertising between zones;
- external1 – OSPF format 1 external routes advertising;
- external2 – OSPF format 2 external routes advertising;

<NAME> – name of the route map that will be used for advertised OSFP routes filtration and modification, set by the string of up to 31 characters.

<LEVEL> — IS-IS protocol level into which routes are advertised (only relevant in CONFIG-ISIS configuration mode):

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2;

Required privilege level

10

Command mode

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-OSPFV3

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

Example

```
esr(config-bgp-af)# redistribute ospf 10 external2
```

redistribute rip

This command enables advertising of routes from the RIP route database.

The use of a negative form (no) of the command disables advertising of routes from the RIP route database.

Syntax

```
redistribute rip [ route-map <NAME> ] [ <LEVEL> ]
```

```
no redistribute rip
```

Parameters

<NAME> – name of the route map that will be used for advertised RIP routes filtration and modification, set by the string of up to 31 characters.

<LEVEL> — IS-IS protocol level into which routes are advertised (only relevant in CONFIG-ISIS configuration mode):

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2;

Required privilege level

10

Command mode

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-OSPFV3

CONFIG-ISIS

Example

```
esr(config-bgp)# redistribute rip
```

redistribute static

This command enables static routes advertising.

The use of a negative form (no) of the command disables static routes advertising.

Syntax

```
redistribute static [ route-map <NAME> ] [ <LEVEL> ]
no redistribute static
```

Parameters

<NAME> – name of the route map that will be used for advertised static routes filtration and modification, set by the string of up to 31 characters.

<LEVEL> — IS-IS protocol level into which routes are advertised (only relevant in CONFIG-ISIS configuration mode):

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2;

Required privilege level

10

Command mode

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-OSPFV3

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

Example

```
esr(config-bgp)# redistribute static
```

Policy-based routing (PBR)

- action
- action set as-path prepend
- action set community
- action set extcommunity
- action set ip bgp-next-hop
- action set ip next-hop
- action set ip next-hop verify-availability
- action set ipv6 bgp-next-hop
- action set ipv6 next-hop
- action set local-preference
- action set metric bgp
- action set metric isis
- action set metric ospf
- action set metric rip
- action set origin
- action set tag ospf
- action set tag rip
- action set weight bgp
- description
- ip policy route-map
- match as-path
- match community
- match extcommunity
- match ip access-group
- match ip address
- match ip address object-group
- match ip next-hop
- match ip route-source
- match ipv6 address
- match ipv6 address object-group
- match ipv6 next-hop
- match ipv6 route-source
- match metric bgp
- match metric ospf
- match metric rip
- match tag ospf
- match tag rip
- match weight bgp
- route-map
- route-map
- rule
- show ip route-map

action

The command specifies the action that should be applied for the route information meeting this requirements. It is used only for filtering routes of dynamic routing protocols and has no effect when configuring PBR.

The use of a negative form (no) of the command sets the default value.

Syntax

```
action <ACT>
```

```
no action
```

Parameters

<ACT> – allocated action:

- permit – routing information reception/advertising is permitted;
- deny – routing information reception/advertising is denied;

Default value

permit

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action deny
```

action set as-path prepend

This command sets BGP AS-Path attribute value that will be added to the beginning of AS-Path list (optionally).

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set as-path prepend <AS-PATH> { track <TRACK-ID> }
```

```
no set as-path
```

Parameters

<AS-PATH> – autonomous systems number list that will be added to the current value in the route. It is defined as AS,AS,AS where each part takes values of [1..4294967295]. You can specify up to 10 autonomous system numbers.

<TRACK-ID> – the tracking identifier of the object, if all the conditions for which the specified action is executed. Changes in the range of [1..60].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set as-path prepend 100,200,300
```

action set community

This command sets the value of the BGP Community attribute to be set in the route.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set community { <COMMUNITY-LIST> | no-advertise | no-export }
no action set community
```

Parameters

<COMMUNITY-LIST> – community list, defined as AS:N,AS:N,AS:N, where AS part takes values of [1..65535] and N part takes values of [0..65535]. You can specify up to 64 community;

no-advertise – when specifying a command, routes that are transmitted with this community attribute value should not be advertised to other BGP neighbors;

no-export — when specifying a command, routes that are transmitted with this community attribute value should not be advertised outside the confederation (an autonomous system that is not part of the confederation is considered a confederation). That is, routes are not advertised to eBGP neighbors, but are advertised by external neighbors in the confederation.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set community no-advertise
```

action set extcommunity

This command sets the value of the BGP ExtCommunity attribute to be set in the route.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set extcommunity <EXTCOMMUNITY-LIST>
```

```
no action set extcommunity
```

Parameters

<EXTCOMMUNITY-LIST> – community list, defined as KIND:AS:N,KIND:AS:N,KIND:AS:N, where

- KIND – extcommunity type, take rt (Route Target) and ro (Route Origin) values;
- <AS> – autonomous system number, takes values of [1..4294967295];
- N – extcommunity number, which defines the traffic routing policy, takes the values [1..65535].

You can specify up to 64 ExtCommunity.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set extcommunity ro:435:6
```

action set ip bgp-next-hop

This command sets the value of the BGP Next-Hop attribute to be set in the route when BGP is advertised.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set ip bgp-next-hop <ADDR>
```

```
no action set ip next-hop
```

Parameters

<ADDR> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set ip bgp-next-hop 10.100.100.1
```

action set ip next-hop

This command specifies Next-Hop value that will be set in the route received by BGP.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set ip next-hop { <NEXTHOP> | blackhole | unreachable | prohibit }
no action set ip next-hop
```

Parameters

<NEXTHOP> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

- blackhole – when specifying the command, the packets to this subnet will be removed by the device without sending notifications to a sender;
- unreachable – when specifying the command, the packets to this subnet will be removed by the device, a sender will receive in response ICMP Destination unreachable (Host unreachable, code 1);
- prohibit – when specifying the command, the packets to this subnet will be removed by the device, a sender will receive in response ICMP Destination unreachable (Communication administratively prohibited, code 13).

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set ip next-hop prohibit
```

action set ip next-hop verify-availability

This command sets Next-Hop for the packets that meet the requirements of the specified ACL.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set ip next-hop verify-availability <NEXTHOP> <METRIC>
no action set ip next-hop verify-availability { <NEXTHOP> | all }
```

Parameters

<NEXTHOP> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<METRIC> – route metric, takes values of [1..255].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set ip next-hop verify-availability 1.1.1.1 25
```

action set ipv6 bgp-next-hop

This command sets the value of the BGP Next-Hop attribute for IPv6 to be set in the route when BGP is advertised.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set ipv6 bgp-next-hop <ADDR>
no action set ipv6 next-hop
```

Parameters

<IPV6-ADDR> – gateway IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set ipv6 bgp-next-hop 2002::765:1
```

action set ipv6 next-hop

This command specifies Next-Hop value for IPv6 that will be set in the route received by BGP.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set ipv6 next-hop <NEXTHOP>
no action set ipv6 next-hop
```

Parameters

<IPV6-ADDR> – client IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set ipv6 next-hop 55::205:2
```

action set local-preference

This command sets the value of the BGP Local Preference attribute to be set in the route.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set local-preference <PREFERENCE>
no action set local-preference
```

Parameters

<PREFERENCE> – BGP Local Preference attribute value, takes values in the range of [1..2147483647].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set local-preference 120
```

action set metric bgp

This command sets the value of the BGP MED attribute to be set in the route.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set metric bgp <METRIC> { track <TRACK-ID> }
no action set metric bgp
```

Parameters

<METRIC> – BGP MED attribute value, takes values in the range of [0..4294967295].

<TRACK-ID> – the tracking identifier of the object, if all the conditions for which the specified action is executed. Changes in the range of [1..60].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set metric bgp 10
```

action set metric isis

This command sets the value of the IS-IS Metric attribute to be set in the route.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set metric isis <METRIC>
no action set metric isis
```

Parameters

<METRIC> – IS-IS Metric attribute value, may take values [1..16777215].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set metric isis 10
```

action set metric ospf

This command sets the value of the OSPF Metric attribute to be set in the route.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set metric ospf <TYPE> <METRIC>
no action set metric ospf
```

Parameters

<TYPE> – OSPF Metric attribute type, takes values type-1 and type-2.

<METRIC> – OSPF Metric attribute value, takes values in the range of [0..65535].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set metric ospf type-1 10
```

action set metric rip

This command sets the value of the RIP Metric attribute to be set in the route.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set metric rip <METRIC>
```

```
no action set metric rip
```

Parameters

<METRIC> – RIP Metric attribute value, takes values in the range of [0..16].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set metric rip 5
```

action set origin

This command sets the value of the BGP Origin attribute to be set in the route.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set origin <ORIGIN>
no action set origin
```

Parameters

<ORIGIN> – value of the BGP Origin attribute, takes the following values:

- egp – route is learnt by Exterior Gateway Protocol (EGP);
- igp – route is received inside the initial AS;
- incomplete – route is learnt in another way.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set origin igp
```

action set tag ospf

This command sets the value of the OSPF Tag attribute to be set in the route.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set tag ospf <TAG>
no action set tag ospf
```

Parameters

<TAG> – OSPF Tag attribute value, takes values in the range of [0..4294967295].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set tag ospf 20
```

action set tag rip

This command sets the value of the RIP Tag attribute to be set in the route.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set tag rip <RIP>
no action set tag rip
```

Parameters

<RIP> – RIP Tag attribute value, takes values in the range of [0..65535].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set tag rip 20
```

action set weight bgp

This command sets the value of the BGP weight attribute to be set in the route.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
action set weight bgp <WEIGHT>
no action set weight bgp
```

Parameters

<WEIGHT> – value of the BGP weight attribute takes the values [0..65535].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# action set weight bgp 200
```

description

This command changes a description of configurable route map rule.

The use of a negative form (no) of the command removes a specified description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – route map rule description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# description "Drop Local NETs"
```

ip policy route-map

This command assigns a routing policy based on access lists (ACL) to interface.

The use of a negative form (no) of the command removes routing policy.

Syntax

```
ip policy route-map <NAME>
no ip policy route-map
```

Parameters

<NAME> – configured routing policy name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-CELLULAR-MODEM

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

Example

```
esr(config-subif)# route-map drop-local-net in
```

match as-path

This command sets BGP AS-Path attribute value in the route for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match as-path [ begin | end | contain ] <AS-PATH>
```

```
no match as-path
```

Parameters

<AS-PATH> – list of autonomous system numbers, defined as AS,AS,AS, where every part takes values of [1..4294967295]. You can specify up to 10 autonomous system numbers.

Optional parameters with the use of which the partial correspondence of the attribute is specified:

- begin – attribute value begins with the specified AS numbers list;
- end – attribute value ends with the specified AS numbers list;
- contain – attribute value includes the specified AS numbers list.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match as-path begin 100,200,300
```

match community

This command sets BGP Community attribute value in the route for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match community <COMMUNITY-LIST>
```

```
no match community
```

Parameters

<COMMUNITY-LIST> – community list, defined as AS:N,AS:,AS, where AS part takes values of [1..65535] and N part takes values of [0..65535]. You can specify up to 64 community.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match community 100:1,200:3,300:65000
```

match extcommunity

This command sets BGP ExtCommunity attribute value in the route for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match extcommunity <EXTCOMMUNITY-LIST>
no match extcommunity
```

Parameters

<EXTCOMMUNITY-LIST> – extcommunity list, defined as KIND:AS:N, KIND:AS:N, KIND:AS:N, where:

- KIND – extcommunity type, take rt (Route Target) or ro (Route Origin) values;
- <AS> – autonomous system number, takes values of [1..4294967295];
- N – extcommunity number, which defines the traffic routing policy, takes the values [1..65535].

You can specify up to 64 extcommunity.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match extcommunity ro:435:6
```

match ip access-group

The command sets ACL group for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match ip access-group <NAME>
no match ip access-group
```

Parameters

<NAME> – access control list name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match ip access-group ACCESS
```

match ip address

The command sets route addresses for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match ip address <ADDR/LEN> [ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ] }
no match ip address prefix
```

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<LEN> – prefix length, takes values [1..32];

eq – when specifying the command, the prefix length mast match the specified one;

le – when specifying the command, the prefix length mast be less than or match the specified one;

ge – when specifying the command, the prefix length mast be more than or match the specified one.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match ip address 192.168.0.0/32 ge 16
```

match ip address object-group

This command sets an IP address profile containing the values of destination subnets in the route for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match ip address object-group <OBJ-GROUP- NETWORK -NAME> [ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ]
```

```
no match ip address object-group
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes destination subnets prefixes, set by the string of up to 31 characters;

eq – when specifying the command, the prefix length mast match the specified one;

le – when specifying the command, the prefix length mast be less than or match the specified one;

ge – when specifying the command, the prefix length mast be more than or match the specified one.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match ip address object-group local_nets
```

match ip next-hop

This command sets an IP address profile containing the values of BGP Next-Hop attribute in the route for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match ip next-hop object-group <OBJ-GROUP-NETWORK-NAME>
```

```
no match ip next-hop
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes gateway IP address ranges, set by the string of up to 31 characters;

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match ip next-hop object-group block_nexthop
```

match ip route-source

This command sets IP addresses profile. Profile contains IP addresses of the router that advertised the route. Used to filter by source IP address when advertising route information.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match ip route-source object-group <OBJ-GROUP-NETWORK-NAME>
no match ip route-source
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes IP address ranges of the route information source, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match ip route-source object-group source_routers
```

match ipv6 address

The command sets IPv6 route addresses for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match ipv6 address <IPV6-ADDR/LEN> [ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ] }
```

```
no match ipv6 address prefix
```

Parameters

<IPV6-ADDR> – IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<LEN> – prefix length, takes values [1..128];

eq – when specifying the command, the prefix length mast match the specified one;

le – when specifying the command, the prefix length mast be less than or match the specified one;

ge – when specifying the command, the prefix length mast be more than or match the specified one.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match ipv6 address FF02:0:0:0:0:1:FF00::/104
```

match ipv6 address object-group

This command sets an IPv6 address profile containing the values of destination subnets in the route for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match ipv6 address object-group <OBJ-GROUP- NETWORK -NAME> [ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ]
```

```
no match ipv6 address object-group
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – name of the IPv6 addresses profile that includes destination subnets prefixes, set by the string of up to 31 characters;

eq – when specifying the command, the prefix length mast match the specified one;

le – when specifying the command, the prefix length mast be less than or match the specified one;

ge – when specifying the command, the prefix length mast be more than or match the specified one.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match ipv6 address object-group local_nets
```

match ipv6 next-hop

This command sets an IPv6 address profile containing the values of BGP Next-Hop attribute in the route for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match ipv6 next-hop object-group <OBJ-GROUP-NETWORK-NAME>
no match ipv6 next-hop
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – name of the IPv6 addresses profile that includes gateway IPv6 address ranges, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match ipv6 next-hop object-group block_nexthop
```

match ipv6 route-source

This command sets IPv6 addresses profile. Profile contains IPv6 addresses of the router that advertised the route. Used to filter by source IPv6 address when advertising route information.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match ipv6 route-source object-group <OBJ-GROUP-NETWORK-NAME>
no match ipv6 route-source
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – name of the IPv6 addresses profile that includes IPv6 address ranges of the route information source, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match ipv6 route-source object-group source_routers
```

match metric bgp

This command sets BGP MED attribute value in the route for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match metric bgp <METRIC>
no match metric bgp
```

Parameters

<METRIC> – BGP MED attribute value, takes values in the range of [0..4294967295].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match metric bgp 10
```

match metric ospf

This command sets OSPF Metric attribute value in the route for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match metric ospf <TYPE> <METRIC>
no match metric ospf
```

Parameters

<TYPE> – OSPF Metric attribute type, takes values type-1 and type-2.

<METRIC> – OSPF Metric attribute value, takes values in the range of [0..65535].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match metric ospf type-1 10
```

match metric rip

This command sets RIP Metric attribute value in the route for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match metric rip <METRIC>
no match metric rip
```

Parameters

<METRIC> – RIP Metric attribute value, takes values in the range of [0..16].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match metric rip 5
```

match tag ospf

This command sets OSPF Tag attribute value in the route for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match tag ospf <TAG>
no match tag ospf
```

Parameters

<TAG> – OSPF Tag attribute value, takes values in the range of [0..4294967295].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match tag ospf 20
```

match tag rip

This command sets RIP Tag attribute value in the route for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match set tag rip <RIP>
no match set tag rip
```

Parameters

<RIP> – RIP Tag attribute value, takes values in the range of [0..65535].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match tag rip 20
```

match weight bgp

This command sets BGP weight attribute value in the route for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match weight bgp <WEIGHT>
no match weight bgp
```

Parameters

<WEIGHT> – value of the BGP weight attribute takes the values [0..65535].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP-RULE

Example

```
esr(config-route-map-rule)# match bgp weight 20
```

route-map

This command adds filtration and modification of routes in incoming or outgoing directions.

The use of a negative form (no) of the command disables filtration and modification of routes in corresponding direction.

Syntax

```
route-map <NAME> <DIRECTION>
no route-map <DIRECTION>
```

Parameters

<NAME> – configured route map name, set by the string of up to 31 characters;

<DIRECTION> – direction:

- in – filtration and modification of received routes;
- out – filtration and modification of advertised routes.

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

CONFIG-OSPF

CONFIG-OSPFV3

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPND

Example

```
esr(config-bgp-neighbor)# route-map drop-local-net in
```

route-map

This command creates a route map, which will later be used for filtration and modification of the IP routes that are being advertised and received, and it switches to the settings mode of the route map.

The use of a negative form (no) of the command removes a route map.

Syntax

```
[no] route-map <NAME>
```

Parameters

<NAME> – configuring route map name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# route-map drop-local-net
esr(config-route-map)#
```

rule

This command creates route map rule with specified number and switch to the rule parameters configuration mode. The rules are proceeded by the device in number ascending order.

The use of a negative form of the command (no) removes the rule by number or all rules.

Syntax

`[no] rule <ORDER>`

Parameters

<ORDER> – rule number, takes values of [1..10000].

Required privilege level

10

Command mode

CONFIG-ROUTE-MAP

Example

```
esr(config-route-map)# rule 2
esr(config-route-map-rule)#
```

show ip route-map

This command shows route maps.

Syntax

`show ip route-map <NAME> [ <ORDER> ]`

Parameters

<NAME> – router map name, set by the string of up to 31 characters;

<ORDER> – rule number, takes values of [1..10000]. When specifying a rule number, only the given rules information will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip route-map drop-local-net
Order:                                      2
Description:                                Drop route to private nets
Matching pattern:
    Access group                                --
    AS path                                     --
    Community                                   --
    Extcommunity                                --
    BGP metric (MED):                           --
    Address (object-group):                     local_net
    Next hop (object-group):                    --
    Route source (object-group):                --
    RIP metric                                  --
    RIP tag                                     --
    OSPF metric type                            --
    OSPF metric                                 --
    OSPF tag                                    --
Actions:
    Decision:                                   Deny
    Route next hop address:                     --
    Route IPv6 next hop address:                --
    Route next hop:                             --
    AS path (prepand):                          --
    Community:                                  --
    Extcommunity:                               --
    Local preference:                           --
    BGP next hop address:                       --
    BGP IPv6 next hop address:                  --
    BGP metric (MED):                           --
    Origin:                                     --
    RIP metric                                  --
    RIP tag                                     --
    OSPF metric type                            --
    OSPF metric                                 --
    OSPF tag                                    --
```

Key bindings configuration

- accept-lifetime
- key
- key-chain
- key-string
- send-lifetime

accept-lifetime

This command defines the period of time during which this key can be used to authenticate received packets.

The use of a negative form (no) of the command sets the default value.

Syntax

```
accept-lifetime <TIME_B> <DAY_B> <MONTH_B> <YEAR_B> <TIME_E> <DAY_E> <MONTH_E> <YEAR_E>
no accept-lifetime
```

Parameters

<TIME_B> – set time to start using the key, is given as HH: MM: SS, where:

- HH – hours, takes the value of [0..23];
- MM – minutes, takes the value of [0 ..59];
- SS – seconds, takes the value of [0..59];

<DAY_B> – the day of the month when the key was first used, takes the values [1..31];

<MONTH_B> – month of the start of key usage, takes values:

- January;
- February;
- March;
- April;
- May;
- June;
- July;
- August;
- September;
- October;
- November;
- December;

<YEAR_B> – the year when the key was first used, takes the values [2001..2037].

<TIME_E> – set time to end using the key, is given as HH: MM: SS, where:

- HH – hours, takes the value of [0..23];
- MM – minutes, takes the value of [0 ..59];
- SS – seconds, takes the value of [0..59];

<DAY_E> – the day of the month when the key is ending, takes the values [1..31];

<MONTH_E> – month of the end of key usage, takes values:

- January;
- February;
- March;

- April;
- May;
- June;
- July;
- August;
- September;
- October;
- November;
- December;

<YEAR_E> – the year when the key is ending, takes the values [2001..2037].

Default value

The key is valid constantly.

Required privilege level

15

Command mode

CONFIG-KEYCHAIN-KEY

Example

```
esr(config-keychain-key)# accept-lifetime 16:35:00 10 May 2015 16:35:00 10 June 2021
```

key

The command adds a key to the key chain and switches to its configuration mode.

The use of a negative form (no) of the command removes a specified key.

Syntax

```
[no] key <ID>
```

Parameters

<ID> – key identifier, set in the range of [0..255].

Required privilege level

15

Command mode

CONFIG-KEYCHAIN

Example

```
esr(config-keychain)# key 25
```

key-chain

The command adds a key chain to the system and switches to its configuration mode.

The use of a negative form (no) of the command removes a specified list.

Syntax

```
[no] key-chain <KEYCHAIN>
```

Parameters

<KEYCHAIN> – key list identifier, set by the string of up to 16 ASCII characters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# key-chain lock
```

key-string

The command sets an authentication password.

The use of a negative form (no) of the command removes the password.

Syntax

```
key-string ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no key-string
```

Parameters

<CLEAR-TEXT> – password of 8 to 16 bytes;
<ENCRYPTED-TEXT> – encrypted password of 8 to 16 bytes (from 16 to 32 characters) in hexadecimal format (0xYYYY ...) or (YYYY ...).

Required privilege level

15

Command mode

CONFIG-KEYCHAIN-KEY

Example

```
esr(config-keychain-key)# key-string ascii-text 123456789
esr(config-keychain-key)# key-string ascii-text encrypted CDE65039E5591FA3F1
```

send-lifetime

This command defines the period of time during which this key can be used to authenticate when sending packets.

The use of a negative form (no) of the command sets the default value.

Syntax

`send-lifetime <TIME_B> <DAY_B> <MONTH_B> <YEAR_B> <TIME_E> <DAY_E> <MONTH_E> <YEAR_E>`

`no send-lifetime`

Parameters

<TIME_B> – set time to start using the key, is given as HH: MM: SS, where:

- HH – hours, takes the value of [0..23];
- MM – minutes, takes the value of [0 ..59];
- SS – seconds, takes the value of [0..59];

<DAY_B> – the day of the month when the key was first used, takes the values [1..31];

<MONTH_B> – the month when the key was first used, takes the values [January/February/March/April/May/June/July/August/September/October/November/December];

<YEAR_B> – the year when the key was first used, takes the values [2001..2037];

<TIME_E> – set time to end using the key, is given as HH: MM: SS, where:

- HH – hours, takes the value of [0..23];
- MM – minutes, takes the value of [0 ..59];
- SS – seconds, takes the value of [0..59];

<DAY_E> – the day of the month when the key is ending, takes the values [1..31];

<MONTH_E> – the month when the key is ending, takes the values [January/February/March/April/May/June/July/August/September/October/November/December];

<YEAR_E> – the year when the key is ending, takes the values [2001..2037].

Default value

The key is valid constantly.

Required privilege level

15

Command mode

CONFIG-KEYCHAIN-KEY

Example

```
esr(config-keychain-key)# send-lifetime 16:35:00 15 May 2014 16:35:00 21 June 2018
```

Event tracking object configuration

- enable
- show ipv6 tracking objects
- show tracking objects
- tracking
- vrrp

enable

This command enables Tracking object.

The use of a negative form (no) of the command disables Tracking object.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

Object disabled.

Required privilege level

10

Command mode

CONFIG-TRACKING

Example

```
esr(config-tracking)# enable
```

show ipv6 tracking objects

This command displays the current status of the tracking objects for the IPv6 stack.

Syntax

```
show ipv6 tracking objects [ <ID> ]
```

Parameters

<ID> – Tracking object number, takes values of [1..60].

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 tracking objects
Track 1 - INACTIVE
```

Example 2

```
esr# show ipv6 tracking objects 1
Track 1 isn't in active condition
Track 1:
    VRRP 2 condition state MASTER
Tracked by:
```

show tracking objects

This command displays the current status of the tracking objects for the IPv4 stack.

Syntax

```
show tracking objects [ <ID> ]
```

Parameters

<ID> – Tracking object number, takes values of [1..60].

Required privilege level

1

Command mode

ROOT

Example 1

```
esr# show tracking objects
Track 1 - INACTIVE
```

Example 2

```
esr# show tracking objects 1
Track 1 isn't in active condition
Track 1:
    VRRP 2 condition state MASTER
Tracked by:
```

tracking

This command adds Tracking object to the system and switch to the Tracking object parameters configuration mode. The object defines the events to be monitored. When configured events occur, static routes linked to the object are affected. If all conditions are met, the route is added to the system; otherwise, if at least one condition is not met, the route is removed from the system.

The use of a negative form (no) of the command removes Tracking object from the system.

Syntax

`[no] tracking <ID>`

Parameters

<ID> – Tracking object number, takes values of [1..60].

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# tracking 20
esr(config-trackin)#
```

vrrp

This command specifies a rule for keeping track of VRRP process status.

When using 'not' command, the rule will work for all VRRP process statuses except a specified one. The use of a negative form (no) of the command cancels set action.

Syntax

`vrrp <VRID> [not] state { master | backup | fault }`

`no vrrp <VRID>`

Parameters

<VRID> – trackable VRRP router identifier, takes values in the range of [1..255].

Required privilege level

10

Command mode

CONFIG-TRACKING

Example

```
esr(config-tracking)# vrrp 2 state master
```

BFD configuration

- bfd-enable
- ip bfd idle-tx-interval
- ip bfd log-adjacency-changes
- ip bfd min-rx-interval
- ip bfd min-tx-interval
- ip bfd multiplier
- ip bfd neighbor
- ip bfd passive
- ip ospf bfd-enable
- ipv6 bfd log-adjacency-changes
- ipv6 bfd neighbor
- ipv6 ospf bfd-enable
- show ip bfd

bfd-enable

This command enables BFD protocol on the BGP neighbor being configured.

The use of a negative form (no) of the command disables BFD protocol.

Syntax

```
[no] bfd-enable
```

Parameters

The command does not contain parameters.

Default value

Process disabled.

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Example

```
esr(config-bgp-neighbor)# bfd-enable
```

ip bfd idle-tx-interval

This command sets the interval after which the BFD message is sent to the neighbor. This parameter is used to reduce the number of generated messages for those cases when the BFD neighbor is not available or the BFD protocol is turned off.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip bfd idle-tx-interval <TIMEOUT>
```

```
no ip bfd idle-tx-interval
```

Parameters

<TIMEOUT> – interval after which the BFD packet is sent takes a value in milliseconds in the range:

- [200..65535] for ESR-1000/1200/1500/1511/1700/3100;
- [300..65535] for ESR-10/12V/12VF/14VF/20/21/100/200.

Default value

1 second

Required privilege level

10

Command mode

CONFIG

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

Example

```
esr(config)# ip bfd idle-tx-interval 4000
```

ip bfd log-adjacency-changes

This command enables logging of BFD protocol state changes.

The use of a negative form (no) of the command disables logging of BFD protocol state changes.

Syntax

```
ip bfd log-adjacency-changes
no ip bfd log-neighbor-changes
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip bfd log-adjacency-changes
```

ip bfd min-rx-interval

This command sets the minimum interval after which the neighbor should generate BFD message. This parameter is advertising to the neighbor.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip bfd min-rx-interval <TIMEOUT>
no ip bfd min-rx-interval
```

Parameters

<TIMEOUT> – interval after which the BFD message is sent by the neighbor takes a value in milliseconds in the range:

- [200..65535] for ESR-1000/1200/1500/1511/1700/3100;
- [300..65535] for ESR-10/12V/12VF/14VF/20/21/100/200.

Default value

300 ms on ESR-10/12V/12VF/14VF/20/21/100/200

200 ms on ESR-1000/1200/1500/1511/1700/3100

Required privilege level

10

Command mode

CONFIG

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

Example

```
esr(config)# ip bfd min-rx-interval 1000
```

ip bfd min-tx-interval

This command sets the minimal interval after which the BFD message is sent to the neighbor. This parameter is used only when the BFD session is active; in other cases, «p bfd idle-tx-interval is used.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip bfd min-tx-interval <TIMEOUT>
```

```
no ip bfd min-tx-interval
```

Parameters

<TIMEOUT> – interval after which the BFD message is sent to the neighbor takes a value in milliseconds in the range:

- [200..65535] for ESR-1000/1200/1500/1511/1700/3100;
- [300..65535] for ESR-10/12V/12VF/14VF/20/21/100/200.

Default value

300 ms on ESR-10/12V/12VF/14VF/20/21/100/200

200 ms on ESR-1000/1200/1500/1511/1700/3100

Required privilege level

10

Command mode

CONFIG

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

Example

```
esr(config)# ip bfd min-tx-interval 1000
```

ip bfd multiplier

This command sets the amount of dropped packets, at which the BFD neighbor is considered to be unavailable. The time of detection of unavailability in each direction is calculated from the given number multiplied by the tx/rx interval.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip bfd multiplier <COUNT>
no ip multiplier
```

Parameters

<COUNT> – amount of dropped packets, at which the neighbor is considered to be unavailable, takes values in the range of [1..100].

Default value

5

Required privilege level

10

Command mode

CONFIG

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

Example

```
esr(config)# ip bfd multiplier 10
```

ip bfd neighbor

This command puts BFD mechanism with the specified IP address into operation. Operation of this mechanism is necessary to remove static routes from the routing table when next-hop is not available (the bfd key of the ip route command. See section ip route)

The use of a negative form (no) of the command stops mechanism with the specified IP address operation

Syntax

```
ip bfd neighbor <ADDR> [ { interface <IF> | tunnel <TUN> } ] [ local-address <ADDR>
[ multihop ] ] [ vrf <VRF> ]
```

```
no ip bfd neighbor <ADDR> [vrf <VRF>]
```

Parameters

<ADDR> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IF> – an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

<VRF> – VRF name, set by the string of up to 31 characters.

multihop – key for setting TTL=255, for BFD mechanism operation through the routed network.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip bfd neighbor 192.168.0.2
```

ip bfd passive

This command switches BFD session to the passive mode, so that BFD messages will not be sent until the messages from BFD neighbor are received.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] ip bfd passive
```

Parameters

The command does not contain parameters.

Default value

Active mode.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

Example

```
esr(config)# ip bfd passive
```

ip ospf bfd-enable

This command enables the BFD protocol for the OSPF protocol on the interface.

The use of a negative form (no) of the command disables the BFD protocol for the OSPF protocol on the interface.

Syntax

```
[no] ip ospf bfd-enable
```

Parameters

The command does not contain parameters.

Default value

BFD protocol for OSPF protocol disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-LOOPBACK

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf bfd-enable
```

ipv6 bfd log-adjacency-changes

This command enables logging of IPv6 BFD protocol state changes.

The use of a negative form (no) of the command disables logging of IPv6 BFD protocol state changes.

Syntax

```
ipv6 bfd log-adjacency-changes
no ipv6 bfd log-adjacency-changes
```

Parameters

The command does not contain parameters.

Default value

Logging disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 bfd log-adjacency-changes
```

ipv6 bfd neighbor

This command puts BFD mechanism with the specified IPv6 address into operation. This mechanism is necessary to remove bgp routes from the routing table if the next-hop is not available.

The use of a negative form (no) of the command stops mechanism with the specified IPv6 address operation

Syntax

```
ipv6 bfd neighbor <IPV6-ADDR> [ interface <IF> ] [ local-address <IPV6-ADDR>
[ multihop ] ] [ vrf <VRF> ]
```

```
no ipv6 bfd neighbor <IPV6-ADDR> [ vrf <VRF> ]
```

Parameters

<IPV6-ADDR> – client IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<IF> – an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

<VRF> – VRF name, set by the string of up to 31 characters.

multihop – key for setting TTL=255, for BFD mechanism operation through the routed network.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 bfd neighbor FF02:0:0:0:0:1:FF00::/104
```

ipv6 ospf bfd-enable

This command enables the BFD protocol for the OSPFv3 protocol on the interface.

The use of a negative form (no) of the command disables the BFD protocol for the OSPFv3 protocol on the interface.

Syntax

```
[no] ipv6 ospf bfd-enable
```

Parameters

The command does not contain parameters.

Default value

BFD protocol for OSPF protocol disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf bfd-enable
```

show ip bfd

This command displays information about the BFD protocol parameters or specific interfaces when using a filter.

Syntax

```
show ip bfd [ { interface <IF> | tunnel <TUN> } ]
```

Parameters

<IF> – an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces.

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip bfd
Minimum RX interval: 200 ms
Minimum TX interval: 200 ms
Idle TX interval:    1000 ms
Multiplier:          5 packets
Passive:             No
esr# show ip bfd interface gi1/0/1
Minimum RX interval: 200 ms
Minimum TX interval: 200 ms
Idle TX interval:    1000 ms
Multiplier:          10 packets
Passive:             Yes
```

IPv4/IPv6 static routes configuration

- ip route
- ipv6 route

ip route

This command creates a static IP route to the specified subnet.

The use of a negative form (no) of the command removes a specified route.

Syntax

```
ip route [ vrf <VRF> ] <SUBNET> { { <NEXTHOP> [ resolve ] [ bfd ] | interface <IF> |
tunnel <TUN> | blackhole | unreachable | prohibit } [ track <TRACK-ID> ] [ name <NAME>] }
| wan load-balance rule <RULE> } [ <METRIC> ] no ip route [ vrf <VRF> ] <SUBNET>
[ <METRIC> ]
```

Parameters

<VRF> – VRF name, set by the string of up to 31 characters.

<SUBNET> – destination address, can be specified in the following formats:

- BBB.CCC.DDD – host IP address, where each part takes values of [0..255].
- BBB.CCC.DDD/NN – network IP address with prefix mask, where AAA-DDD take values of [0..255] and NN takes values of [1..32].

<NEXTHOP> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

resolve – when specifying this parameter, gateway IP address will be recursively calculated through the routing table. If the recursive calculation fails to find a gateway from a directly connected subnet, then this route will not be installed into the system.

<IF> – an IP interface name specified in the form described in Section Types and naming order of router interfaces.

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

<RULE> – wan rule number, set in the range of [1..50].

blackhole – when specifying the command, the packets to this subnet will be removed by the device without sending notifications to a sender;

unreachable – when specifying the command, the packets to this subnet will be removed by the device, a sender will receive in response ICMP Destination unreachable (Host unreachable, code 1);

prohibit – when specifying the command, the packets to this subnet will be removed by the device, a sender will receive in response ICMP Destination unreachable (Communication administratively prohibited, code 13);

<METRIC> – route metric, takes values of [0..255].

<TRACK-ID> – Tracking object identifier. If the router is bound to the Tracking object, it will appear in the system only after meeting all requirements specified in the object.

<NAME> – name (description) of the route, text variable up to 31 characters long.

bfd – when specifying this key, the removal of static route in case of next-hop unavailability is activated. For operation of this mechanism, the BFD mechanism must be running with the IP address of the next-hop (see section ip bfd neighbor).

next-hop check using bfd protocol. If next-hop is unavailable, the route is deleted.

⚠ If 0.0.0.0/0 is specified as the subnet, then the default route will be set.

Required privilege level

10

Command mode

CONFIG

Example 1

Set the route to the subnet 192.165.3.0/24 with metric 6 through the gateway 192.165.56.65:

```
esr(config)# ip route 192.165.3.0/24 192.165.56.65 6
```

Example 2

Set the route to the subnet 192.165.3.0/24 with metric 6 through the GigabitEthernet 1/0/5 interface:

```
esr(config)# ip route 192.165.3.0/24 interface gigabitethernet 1/0/5 6
```

Example 3

Set the route to the subnet 192.165.3.0/24 through the tunnel interface vti 1:

```
esr(config)# ip route 192.165.3.0/24 interface vti 1
```

ipv6 route

This command creates a static IPv6 route to the specified subnet.

The use of a negative form (no) of the command removes a specified route.

Syntax

```
ipv6 route [ vrf <VRF> ] <SUBNET> { { <NEXTHOP> [ resolve ] [bfd] | interface <IF> |
blackhole | unreachable | prohibit [ <METRIC> ] [ name <NAME>] } | wan load-balance rule
<RULE> [ <METRIC> ] }
```

```
no ipv6 route [ vrf <VRF> ] <SUBNET> [ <METRIC> ]
```

Parameters

<VRF> – VRF name, set by the string of up to 31 characters.

<SUBNET> – destination address, can be specified in the following formats:

- The addresses are defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].
- <IPV6-ADDR/LEN> – IP address and mask of a subnet, defined as X:X:X:X::X/EE where each X part takes values in hexadecimal format [0..FFFF] and EE takes values of [1..128].

<IPV6-ADDR> – client IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

resolve – when specifying this parameter, gateway IPv6 address will be recursively calculated through the routing table. If the recursive calculation fails to find a gateway from a directly connected subnet, then this route will not be installed into the system.

<IF> – an IP interface name specified in the form described in Section Types and naming order of router interfaces.

blackhole – when specifying the command, the packets to this subnet will be removed by the device without sending notifications to a sender.

unreachable – when specifying the command, the packets to this subnet will be removed by the device, a sender will receive in response ICMP Destination unreachable (Host unreachable, code 1).

prohibit – when specifying the command, the packets to this subnet will be removed by the device, a sender will receive in response ICMP Destination unreachable (Communication administratively prohibited, code 13).

<METRIC> – route metric, takes values of [0..255].

<NAME> – name (description) of the route, text variable up to 31 characters long.

bfd - when specifying this key, the next-hop check is activated using the bfd protocol. If next-hop is unavailable, the route is deleted.

⚠ If ::/0 is specified as the subnet, then the default route will be set.

Required privilege level

10

Command mode

CONFIG

Example 1

Set the route to the subnet 2001:/120 with metric 6 through the gateway fc00::1:

```
esr(config)# ipv6 route 2001::/120 fc00::1 6
```

Example 2

Set the route to the subnet 2001::/120 with metric 6 through the GigabitEthernet 1/0/5 interface:

```
esr(config)# ipv6 route 2001::/120 interface gigabitethernet 1/0/5  6
```

Example 3

Set the route to the subnet 2001::/120 through the tunnel interface vti 1:

```
esr(config)# ipv6 route 2001::/120 interface vti 1
```

BGP configuration

- address-family
- address-family
- allow-local-as
- clear bgp
- cluster-id
- default-originate
- description
- ebgp-multihop
- flow-spec enable
- ipv6 router bgp log-neighbor-changes
- ipv6 router bgp maximum-paths
- neighbor
- next-hop-self
- peer-group
- peer-group
- preference
- remote-as
- remove-private-as
- router bgp
- router bgp log-neighbor-changes
- router bgp maximum-paths
- route-reflector-client
- router-id
- show ip bgp
- show ip bgp summary
- show ip bgp neighbors
- show ipv6 bgp
- show ipv6 bgp neighbors
- show ipv6 bgp summary
- show ip bgp vpnv4
- show ip bgp l2vpn vpls
- send-community extended
- timers error-wait
- timers holdtime
- timers keepalive
- update-source
- vrf
- weight

address-family

This command enters the configuration mode for the BGP process to configure the protocol information transmission parameters.

The use of a negative form (no) of the command removes the configured protocol information transmission parameters.

Syntax:

```
[no] address-family { ipv4 | ipv6 } unicast
```

Parameters:

- ipv4 – ipv4 family;
- ipv6 – ipv6 family.

Required privilege level:

10

Command mode

CONFIG-BGP

CONFIG-BGP-VRF

Example

```
esr(config-bgp)# address-family ipv4
```

address-family

This command defines the type of configured routing information and the transition to this configuration mode

The use of a negative form (no) of the command removes the configured protocol information transmission parameters.

Syntax:

```
[no] address-family { ipv4 | ipv6 | vpnv4 } unicast
```

Parameters:

- ipv4 – ipv4 family;
- ipv6 – ipv6 family;
- vpnv4 – vpnv4 family;
- l2vpn vpls – l2vpn family.

Required privilege level:

10

Command mode:

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Example

```
esr(config-bgp-neighbor)# address-family ipv4
```

allow-local-as

This command enables the mode in which the reception of routes in the BGP attribute, AS Path of which includes the numbers of process autonomous system, is allowed.

The use of a negative form (no) of the command disables this feature.

Syntax

```
allow-local-as <NUMBER>
no allow-local-as
```

Parameters

<NUMBER> – threshold amount of instances of autonomous system number in the AS Path attribute at which the route will be accepted, the range of acceptable values [1..10].

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Example

```
esr(config-bgp-neighbor)# allow-local-as
```

clear bgp

This command resets all or a specific BGP process.

Syntax

```
clear bgp [ <AS> ] [ vrf <NAME_VRF> ] [ neighbor { <ADDR> | <IPV6_ADDR> } ]
```

Parameters

<AS> – autonomous system number, takes values of [1..4294967295];

<ADDR> – neighbor’s IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – neighbor IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<NAME-VRF> – VRF instance name, set by the string of up to 31 characters, within which the BGP sessions will be reset.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear bgp
esr# clear bgp 1000
```

cluster-id

This command sets the Route-Reflector identifier of the cluster to which the router BGP process belongs.

The use of a negative form (no) of the command removes the identifier.

Syntax

```
cluster-id <ID>
no cluster-id
```

Parameters

<ID> – Route-Reflector cluster identifier, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-BGP

CONFIG-BGP-VRF

Example

```
esr(config-bgp-af)# cluster-id 1.1.1.1
```

default-originate

This command sets the mode in which the default route is always sent to the BGP neighbor in the update along with other routes.

The use of a negative form (no) of the command disables this feature.

Syntax

`[no] default-originate`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMALY

CONFIG-BGP-VRF-NEIGHBOR-FAMALY

Example

```
esr(config-bgp-neighbor)# default-originate
```

description

This command defines neighbor description.

The use of a negative form (no) of the command removes description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – neighbor description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Example

```
esr(config-bgp-neighbor)# description "ISP_RTK"
```

ebgp-multihop

This command allows connections to neighbors that are located not in directly connected subnets.

The use of a negative form (no) of the command disables this feature.

Syntax

`[no] ebgp-multihop <NUM>`

Parameters

<NUM> – maximum amount of hops when installing EBGP (used for TTL).

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-NEIGHBOR

Example

```
esr(config-bgp-neighbor)# ebgp-multihop
```

flow-spec enable

This command sets the mode of broadcasting of flow-spec information when working with a BGP neighbor or a group of BGP neighbors.

The use of a negative form (no) of the command sets the default value.

Syntax

`[no] flow-spec enable`

Parameters

None.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMALY

CONFIG-BGP-VRF-NEIGHBOR-FAMALY

Example

```
esr(config-bgp-group)# flow-spec enable
```

ipv6 router bgp log-neighbor-changes

This command enables logging of IPv6 BGP neighbors state changes.

The use of a negative form (no) of the command disables logging of IPv6 BGP neighbors state changes.

Syntax

```
[no] ipv6 router bgp log-neighbor-changes
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 router bgp log-neighbor-changes
```

ipv6 router bgp maximum-paths

This command enables ECMP and defines the maximum amount of equal routes to a destination point.

The use of a negative form (no) of the command disables ECMP.

Syntax

```
ipv6 router bgp maximum-paths <VALUE>
no ipv6 router bgp maximum-paths
```

Parameters

<VALUE> – amount of valid equal IPv6 routes to the target, takes the values of [1..16].

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 router bgp maximum-paths 14
```

neighbor

This command adds BGP neighbor and switchs to the BGP process parameters configuration mode. The use of a negative form (no) of the command removes neighbor router parameters from configuration.

Syntax

```
[no] neighbor {<ADDR> |<IPV6-ADDR>}
```

Parameters

<ADDR> – neighbor's IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<IPV6-ADDR> – client IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

10

Command mode

CONFIG-BGP

CONFIG-VRF-BGP

Example

```
esr(config-bgp)# neighbor 192.168.0.2
esr(config-bgp-neighbor)#
```

next-hop-self

This command sets the mode in which all updates are sent to BGP neighbor with the IP address of a local router outgoing interface as the next-hop.

The use of a negative form (no) of the command disables this feature.

Syntax

```
[no] next-hop-self
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMALY

CONFIG-BGP-VRF-NEIGHBOR-FAMALY

Example

```
esr(config-bgp-neighbor)# next-hop-self
```

peer-group

This command creates a group of BGP neighbors with the specified name and switches to the configuration mode of the group parameters.

The use of a negative form (no) of the command removes BGP group with the specified name.

Syntax

```
[no] peer-group <NAME>
```

Parameters

<NAME> – group name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-BGP

CONFIG-BGP-VRF

Example

```
esr(config-bgp)# peer-group list1
```

peer-group

This command applies the settings described in the specified BGP group to the BGP group or BGP neighbor.

The use of a negative form (no) of the command cancels applying of the settings described in the specified BGP group

Syntax

```
peer-group <NAME>
no peer-group
```

Parameters

<NAME> – group name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Example

```
esr(config-bgp-af)# peer-group list1
```

preference

This command defines the precedence of the routes received from a neighbor.

The use of a negative form (no) of the command sets the default value.

Syntax

```
preference <VALUE>
no preference
```

Parameters

<VALUE> – neighbor routes precedence, takes values in the range of [1..255].

Default value

170

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMALY

CONFIG-BGP-VRF-NEIGHBOR-FAMALY

Example

```
esr(config-bgp-neighbor)# preference 30
```

remote-as

This command sets the number of BGP neighbor autonomous system.

The use of a negative form (no) of the command removes a number of autonomous system.

Syntax

```
remote-as <AS>
no remote-as
```

Parameters

<AS> – autonomous system number, takes values of [1..4294967295].

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Example

```
esr(config-bgp-neighbor)# remote-as 20
```

remove-private-as

This command sets the mode in which private numbers of autonomous systems are removed from the AS Path routes BGP attribute before sending an update (in accordance with RFC 6996).

The use of a negative form (no) of the command disables this feature.

Syntax

```
[no] remove-private-as <ACTION>
```

Parameters

<ACTION> – an action indicating which private AS should be removed or replaced. One of the following:

- all – remove all private AS inclusions from the original AS path;
- nearest – delete all private AS to the last (right) public AS in the original AS path;
- replace – replace all private AS with the AS number during which this command is processed.

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

Example

```
esr(config-bgp-neighbor-af)# remove-private-as
```

router bgp

This command adds BGP process to the system and switchs to the BGP process parameters configuration mode.

The use of a negative form (no) of the command removes a BGP process from the system.

Syntax

```
[no] router bgp <AS>
```

Parameters

<AS> – autonomous system number, takes values of [1..4294967295].

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# router bgp 1000
esr(config-bgp)#
```

Added BGP process with autonomous system 1000.

router bgp log-neighbor-changes

This command enables logging of BGP neighbors state changes.

The use of a negative form (no) of the command disables logging of BGP neighbors state changes.

Syntax

```
[no] router bgp log-neighbor-changes
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# router bgp log-neighbor-changes
```

router bgp maximum-paths

This command enables ECMP and defines the maximum amount of equal routes to a destination point.

The use of a negative form (no) of the command disables ECMP.

Syntax

```
router bgp maximum-paths <VALUE>
no router bgp maximum-paths
```

Parameters

<VALUE> – amount of valid equal routes to the target, takes the values of [1..16].

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# router bgp maximum-paths 14
```

route-reflector-client

This command specifies BGP neighbor as a Route-Reflector client.

The use of a negative form (no) of the command disables this feature.

Syntax

```
[no] route-reflector-client
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Example

```
esr(config-bgp-neighbor)# route-reflector-client
```

router-id

This command sets router identifier.

The use of a negative form (no) of the command removes the identifier.

Syntax

```
router-id <ID>
no router-id
```

Parameters

<ID> – router identifier, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-BGP

CONFIG-BGP-VRF

Example

```
esr(config-bgp-af)# router-id 1.1.1.1
```

show ip bgp

This command displays the BGP routing table or detailed information about a specific route when using filters.

Syntax

```
show ip bgp [ <AS> [ vrf <VRF> ] { <ADDR> | <ADDR/LEN> ] [flow-spec]
```

Parameters

<AS> – autonomous system number, takes values of [1..4294967295].

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the routing protocol will operate.

<ADDR> – destination IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<ADDR/LEN> – IP subnet, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32].

flow-spec – output flow-spec information from all BGP neighbors.

Required privilege level

1

Command mode

ROOT

Example 1

```
esr# show ip bgp
Terra# show ip bgp
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
              * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete
     Network                Next Hop               Metric  LocPrf  Path
*> u 10.0.10.0/24           10.115.0.1                     100      i
*> u 0.0.0.0/0              10.115.0.1                     100      i
*  u 14.0.10.0/24           10.115.0.1                     100      i
```

Example 2

```
Sword# show ip bgp 75.0.0.0
75.0.0.0/24        via 115.0.0.40 on gi1/0/14        [bgp20 2000-01-15] (AS90?)
    Administrative Distance: 68
    Type:                    unicast
    Origin:                  Incomplete
    AS PATH:                 1 30 70 90
    Next Hop:                115.0.0.40
    MED:                     0
    Local Preference:        100
    Community:               (1:555)
    Valid, Best
```

show ip bgp summary

This command displays information about the status of connections with BGP neighbors.

Syntax

```
show ip bgp <AS> [ vrf <VRF> ] summary
```

Parameters

<AS> – autonomous system number, takes values of [1..4294967295].

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the routing protocol will operate.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip bgp show ip bgp 65054 summary
Mon March 06 08:42:22 2017
  BGP router identifier 10.0.0.1, local AS number 65054
  BGP activity 635190/1270406 prefixes
  Neighbor                 AS              MsgRcvd      MsgSent      Up/Down     St/PfxRcd
  ----------------------   -------------   ----------   ----------   ----------   -------
  95.156.65.5              12389           366960       2260         16:38:50     635189
  100.100.100.1            65535           0            0            911:24:27    Connect
  123.1.1.200              5448            14224        3673129      207:25:28    0
  192.168.16.3             65530           1003         1145         16:38:48     1
  192.168.16.4             65529           0            0            16:38:56     Connect
  192.168.16.100           48858           0            0            911:24:27    Connect
  192.168.16.147           65111           17           120065       00:12:13     0
  192.168.17.18            21127           0            0            16:38:56     Connect
  192.168.17.114           200             0            0            16:38:56     Connect
  192.168.17.155           65500           0            0            16:38:56     Connect
  192.168.25.124           59831           0            0            16:38:56     Active
  200.0.0.1                65200           0            0            16:38:29     IdleAS4
```

show ip bgp neighbors

This command displays information about all or specified BGP neighbor.

Syntax

```
show ip bgp <AS> [ vrf <VRF> ] neighbors [ <ADDR> [ routes | advertise-routes ] ]
```

Parameters

<AS> – autonomous system number, takes values of [1..4294967295].

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the routing protocol will operate.

<ADDR> – neighbor’s IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

routes – when specifying a command, route information received from a neighbor is displayed.

advertise-routes – when you specify a command, it displays routing information advertised to a neighbor.

Required privilege level

1

Command mode

ROOT

Example 1

```
esr# show ip bgp 20 neighbors
BGP neighbor is 10.115.0.1
    BGP state:        Established
    Neighbor address: 10.115.0.1
    Neighbor AS:      20
    Neighbor ID:      115.0.0.1
    Neighbor caps:    refresh restart-aware AS4
    Session:          internal multihop AS4
    Source address:   10.115.0.2
    Hold timer:       137/180
    Keepalive timer:  10/60
   Incoming prefix-list: from_ISP
    Outgoing prefix-list: to_ISP
    Incoming route-map:   comingS
    Outgoing route-map:   AS_prepend
    Uptime:               12 s
    BFD address:          192.168.1.2
    BFD state:            Up
    BFD interval:         3.000 s
    BFD timeout:          15.000 s
```

Example 2

```
esr# show ip bgp 20 neighbors 10.115.0.1 routes
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
              * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete
     Network              Next Hop             Metric  LocPrf  Path
*> u 10.0.10.0/24         10.115.0.1                   100      i
*> u 0.0.0.0/0            10.115.0.1                   100      i
*  u 14.0.10.0/24         10.115.0.1                   100      i
```

Example 3

```
esr# show ip bgp 20 neighbors 115.0.0.40 advertise-routes
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
              * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete
     Network              Next Hop             Metric  LocPrf  Path
*> u 1.1.1.0/24           115.0.0.1            215     100     20 i
*> u 1.1.0.0/24           115.0.0.1            215     100     20 i
*> u 2.2.2.0/24           115.0.0.1            215     100     20 i
```

show ipv6 bgp

This command displays the BGP routing table or detailed information about a specific route when using filters.

Syntax

```
show ipv6 bgp [<AS> [vrf <VRF> [<IPV6-ADDR> |<IPV6-ADDR/LEN>] |<IPV6-ADDR> |<IPV6-ADDR/LEN>] [flow-spec]
```

Parameters

<AS> – autonomous system number, takes values of [1..4294967295].

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the routing protocol will operate.

<IPV6-ADDR> – destination IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

<IPV6-ADDR/LEN> – a subnet, defined as X:X:X:X::X/EE where each part takes values in hexadecimal format [0..FFFF] and EE takes values of [1..128].

flow-spec – output flow-spec information from all BGP neighbors.

Required privilege level

1

Command mode

ROOT

Example 1

```
esr#  show ipv6 bgp 20
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
              * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete
     Network                Next Hop                Metric  LocPrf  Path
*> u 222::/120              44:44:44::44            0       100     1    ?
*> u 40::75:0/120           44:44:44::44            0       100     1    ?
*> u 40::77:0/120           44:44:44::44            0       100     1    ?
*  u 2002::7300:0/120       44:44:44::44            0       100     1    ?
```

Example 2

```
Sword# show ipv6 bgp 20 202::7300:0/120
202::7300:0/120    via 2002::7300:a on gi1/0/14      [bgp20 14:43:48] (AS1000e)
    Administrative Distance: 170
    Type:                    unicast
    Origin:                  EGP
    AS PATH:                 1000
    Next Hop:                2002::7300:a (fe80::6666:b3ff:fe06:cb18)
    MED:                     0
    Local Preference:        188
    Valid, Best
```

show ipv6 bgp neighbors

This command displays information about all or specified BGP neighbor.

Syntax

```
show ipv6 bgp <AS> [ vrf<VRF> ] neighbors [ <IPV6-ADDR> [ routes | advertise-routes ] ]
```

Parameters

<AS> – autonomous system number, takes values of [1..4294967295].

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the routing protocol will operate.

<IPV6-ADDR> – neighbor's IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

routes – when specifying a command, route information received from a neighbor is displayed.

advertise-routes – when you specify a command, it displays routing information advertised to a neighbor.

Required privilege level

1

Command mode

ROOT

Example 1

```
esr# show ipv6 bgp 20 neighbors
BGP neighbor is 2002::7300:c
    BGP state:         Established
    Neighbor address: 2002::7300:c
    Neighbor AS:       1
    Neighbor ID:       77.0.0.1
    Neighbor caps:     refresh
    Session:           external
    Source address:    2002::7300:1
    Hold timer:        127/154
    Keepalive timer:   1/34
BGP neighbor is 2002::7300:a
    BGP state:         Established
    Neighbor address: 2002::7300:a
    Neighbor AS:       1000
    Neighbor ID:       10.10.10.10
    Neighbor caps:     refresh AS4
    Session:           external AS4
    Source address:    2002::7300:1
    Hold timer:        157/180
    Keepalive timer:   32/60
```

Example 2

```
esr# show ipv6 bgp 20 neighbors 2002::7300:a routes
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
              * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete
     Network               Next Hop             Metric  LocPrf   Path
*  u 2002::7300:0/120      2002::7300:a         0       188      1000 e
*> u 220::7300:0/120       2002::7300:a         0       188      1000 e
*> u 22::7300:0/120        2002::7300:a         0       188      1000 e
*> u 2002::40:0/120        2002::7300:a         0       188      1000 e
```

Example 3

```
esr#  show ipv6 bgp 20 neighbors 2002::7300:a advertise-routes
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
              * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete
     Network               Next Hop             Metric  LocPrf   Path
*> u 77::77:0/120          2002::7300:1                 100      20   i
*> u 33::33:0/120          2002::7300:1                 100      20   i
*> u 44::44:44/128         2002::7300:1                 100      20   i
*> u 222::/120             44:44:44::44         0       100      20 1 ?
*> u 40::75:0/120          44:44:44::44         0       100      20 1 ?
```

show ipv6 bgp summary

This command displays information about the status of connections with BGP neighbors.

Syntax

```
show ipv6 bgp <AS> [ vrf <VRF> ] summary
```

Parameters

<AS> – autonomous system number, takes values of [1..4294967295].

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the routing protocol will operate.

Required privilege level

1

Command mode

ROOT

Example

```
esr# sh ipv6 bgp 1 summary
Tue May 10 19:12:12 2011
  BGP router identifier 0.0.0.1, local AS number 1
  BGP activity 0/0 prefixes
  Neighbor         AS              MsgRcvd      MsgSent     Up/Down      St/PfxRcd
  ---------------- -------------   ----------   ---------  ----------   -----------
  2a14::2          10001               3           3        00:00:48        0
```

show ip bgp vpnv4

This command displays information on vpnv4 routes. Specifying a prefix in the command displays detailed information about the prefix

Syntax

```
show ip bgp vpnv4 unicast { all | rd <RD> | vrf <NAME_VRF> } [ neighbor <IP> { routes | advertise-routes } ] [ <PREFIX> ]
```

Parameters

all – when the command all is specified, the full vpnv4 route information is displayed;
rd – when the command rd is specified, information about routes with the specified RD is displayed;
vrf — when the command vrf is specified, the vpnv4 route information for the specified VRF is displayed;
routes – when specifying a command, route information received from a neighbor is displayed;
advertise-routes – when you specify a command, it displays routing information advertised to a neighbor;
<RD> – Route distinguisher value, specified in one of the following forms:

- <ASN>:<nn> — where <ASN> may take values [1..65535], nn may take values [1..65535];
- <ADDR>:<nn> — where <ADDR> specified as AAA.BBB.CCC.DDD/EE, AAA-DDD may take values [0..255], nn may take values [1..65535];
- <4ASN>:<nn> — where <4ASN> may take values [1..4294967295], nn may take values [1..65535];
- <NAME_VRF> – VRF instance name, set by the string of up to 31 characters;

<IP> – neighbor IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];
<PREFIX> – prefix, defined as AAA.BBB.CCC.DDD/EE where AAA-DDD takes values of [0..255] and EE takes values of [1..32].

Required privilege level

1

Command mode

ROOT

Example 1

```
esr# sh ip bgp vpnv4 unicast all
Status codes: * - valid, > - best, i - internal, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Codes Route Distinguisher IP Prefix Next hop Metric Label LocPrf Weight Path
----- --------------------- ------------------ --------------- ---------- -------- ----------
------ ----------------
*>i 65000:10001 10.100.106.1/32 6.6.6.6 -- 16 100 0 ?
*> 65000:10002 10.20.0.0/24 -- -- 17 -- -- ?
*> 65000:10002 10.20.1.0/24 -- -- 17 -- -- ?
*>i 65000:10001 10.100.0.4/30 2.2.2.2 -- 16 100 0 i
*> 65000:10002 172.18.0.0/30 -- -- 17 -- -- ?
*>i 65000:10002 10.30.0.0/24 5.5.5.5 -- 17 100 55 i
*>i 65000:10001 10.100.104.0/24 2.2.2.2 -- 16 100 0 i
*> 65000:10001 10.100.105.0/24 -- -- 16 -- -- ?
*>i 65000:10001 10.100.106.0/24 6.6.6.6 -- 16 100 0 ?
*> 65000:10001 203.203.203.203/32 -- -- 16 -- -- ?
*> 65000:10001 202.202.202.202/32 -- -- 16 -- -- ?
*>i 65000:10001 10.100.100.0/24 2.2.2.2 -- 16 100 0 i
*> 65000:10001 10.100.0.0/30 -- -- 16 -- -- ?
*> 65000:10001 10.100.102.0/24 -- -- 16 -- -- ?
```

Example 2

```
esr# sh ip bgp vpnv4 unicast all 10.100.106.0/24
BGP routing table entry for 65000:10001 10.100.106.0/24
Next hop: 6.6.6.6
Label: 16
AS path: --
Origin: Incomplete
Local preference: 100
Extended Community: RT:65000:10001
Weight: 0
```

show ip bgp l2vpn vpls

This command displays information about all received l2vpn routes.

Syntax

```
show ip bgp l2vpn vpls { all | rd <ASN:nn>} [neighbor <ADDR> advertise-routes] [ ve-id
<ID> block-offset <ID> ]
```

Parameters

- all – when the command all is specified, the full l2vpn route information is displayed;
- rd – when the command rd is specified, information about routes with the specified RD is displayed;
- routes – when specifying a command, route information received from a neighbor is displayed;
- advertise-routes – when you specify a command, it displays routing information advertised to a neighbor;

- <RD> – Route distinguisher value, specified in one of the following forms:
  - <ASN>:<nn> — where <ASN> may take values [1..65535], nn may take values [1..65535];
  - <ADDR>:<nn> — where <ADDR> specified as AAA.BBB.CCC.DDD/EE, AAA-DDD may take values [0..255], nn may take values [1..65535];
  - <4ASN>:<nn> — where <4ASN> may take values [1..4294967295], nn may take values [1..65535].
- <IP> – neighbor IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];
- <ID> – take value [1..65535], specifies the vpls edge or block-offset identifier.

Required privilege level

1

Command mode

ROOT

Example

```
ESR-R#  show ip bgp l2vpn vpls all
Status codes: * - valid, > - best, i - internal, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Codes Route Distinguisher   VID   VBO   VBS   Next hop         Metric     LocPrf     Weight Path
----- --------------------- ----- ----- ----- --------------- ---------- ---------- ------
-------------------
*>i   65502:100             3     1     8     192.168.9.9         --         100        0
i
*>    65502:102             2     1     10    --                  --         --         --
*>    65502:100             2     1     10    --                  --         --         --
*>i   65502:102             1     1     10    192.168.7.7         --         100        0
i
*>i   65502:100             1     1     10    192.168.7.7         --         100        0
i
```

send-community extended

Enables sending of the extended community parameter in adverts sent to a neighbor. By default, extended community options are removed from adverts that are sent. For VPNv4/L2VPN correct operation, sending extended community must be enabled.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] send-community extended
```

Parameters

The command does not contain parameters

Required privilege level

10

Default value

Denied.

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

Example

```
esr(config-bgp-neighbor-af)# send-community extended
```

timers error-wait

This command sets the minimum and maximum delay time during which it is forbidden to establish a connection, in order to protect against frequent disconnections.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timers error-wait <TIME1> <TIME2>
no timers error-wait
```

Parameters

<TIME1> – time in seconds, takes values of [1..65535]:

<TIME2> – time in seconds, takes values of [1..65535].

Default value

60 and 300

Required privilege level

10

Command mode

CONFIG-BGP

CONFIG-VRF-BGP

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Example

```
esr(config-bgp-af)# timers error-wait 90 450
```

timers holdtime

This command sets time interval after which the opposing party is considered to be unavailable. The timer starts after establishing a neighborhood relationship and starts counting from 0. The timer is reset when each reply to a keepalive message from the opposite side is received. It is recommended to set the timer value to *3 * keepalive*.

The use of a negative form (no) of the command sets the default value.

Syntax

`timers holdtime <TIME>`

`no timers holdtime`

Parameters

<TIME> – time in seconds, takes values of [1..65535].

Default value

180

Required privilege level

10

Command mode

CONFIG-BGP

CONFIG-VRF-BGP

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Example

```
esr(config-bgp-af)# timers holdtime 360
```

timers keepalive

This command sets the time interval after which the connection with the opposing party will be checked.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timers keepalive <TIME>
no timers keepalive
```

Parameters

<TIME> – time in seconds, takes values of [1..65535].

Default value

60

Required privilege level

10

Command mode

CONFIG-BGP

CONFIG-VRF-BGP

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Example

```
esr(config-bgp-af)# timers keepalive 120
```

update-source

This command defines the IP/IPv6 address of the router that will be used as the source IP/IPv6 address in BGP routing information updates sent.

The use of a negative form (no) of the command removes a specified source IP/IPv6 address.

Syntax

```
update-source { <ADDR> | <IPV6-ADDR> }
no source-address
```

Parameters

<ADDR> – source IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – source IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Example

```
esr(config-bgp-neighbor)# update-source 10.100.100.2
```

vrf

This command creates an instance of the BGP process in the VRF and switches to the BGP process settings mode in the specified VRF

The use of a negative form (no) of the command removes BGP VRF from the system.

Syntax

```
[no] ip vrf <VRF>
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-BGP

Example

```
esr(config-bgp)# vrf xx
esr(config-bgp-vrf)#
```

weight

This command sets the weight of routes received from a given BGP neighbor or group of BGP neighbors.

The use of a negative form (no) of the command sets the default value.

Syntax

```
weight <WEIGHT>
no weight
```

Parameters

<WEIGHT> – the value of the route weight, takes the values [0..65535].

Default value

0

Required privilege level

10

Command mode

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Example

```
esr(config-bgp-neighbor)# weight 200
```

RIP and RIPNG configuration

- authentication key-chain
- clear ip rip
- clear ipv6 rip
- ip rip metric
- ip rip mode
- ip rip neighbor
- ip rip summary-address
- ipv6 rip metric
- ipv6 rip summary-address
- ipv6 router rip
- passive-interface
- passive-interface
- router rip
- show ip rip
- show ipv6 rip
- timers flush
- timers invalid
- timers update

authentication key-chain

The command specifies a list of passwords for authentication via md5 hash algorithm.

The use of a negative form (no) of the command removes a bind to the password list.

Syntax

```
authentication key-chain <KEYCHAIN>
```

```
no authentication key-chain
```

Parameters

<KEYCHAIN> – key list identifier, set by the string of up to 16 characters.

Required privilege level

10

Command mode

CONFIG-RIP

Example

```
esr(config-rip)# authentication key-chain lock
```

clear ip rip

This command deletes the contents of the RIP route database.

Syntax

```
clear ip rip
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear ip rip
```

clear ipv6 rip

This command deletes the contents of the RIPNG route database.

Syntax

```
clear ipv6 rip
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear ipv6 rip
```

ip rip metric

This command sets the metric value on the interface.

The use of a negative form (no) of the command sets the default metric value.

Syntax

```
ip rip metric <VALUE>
no ip rip metric
```

Parameters

<VALUE> – metric value, defines in values [1..15].

Default value

5

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Example

```
esr(config-if-gi)# ip rip metric 11
```

ip rip mode

This command sets the route advertising mode.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip rip mode <MODE>
no ip rip mode
```

Parameters

<MODE> – route advertising mode:

- multicast – routes are advertised in multicast mode;
- broadcast – routes are advertised in broadcast mode;
- unicast – routes are advertised in unicast mode to neighbors configured using the ***ip rip neighbor <ADDR>***

Default value

multicast

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Example

```
esr(config-if-gi)# ip rip mode broadcast
```

ip rip neighbor

This command statically sets the IP address of the neighbor to establish a relationship in unicast advertising routes mode.

The use of a negative form (no) of the command removes a statically specified neighbor address.

Syntax

```
[no] ip rip neighbor <ADDR>
```

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Example

```
esr(config-if-gi)# ip rip neighbor 10.100.100.5
```

ip rip summary-address

This command enables subnetting.

The use of a negative form (no) of the command disables subnetting.

Syntax

```
[no] ip rip summary-address <ADDR/LEN>
```

Parameters

<ADDR/LEN> – IP subnet, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Example

```
esr(config-if-gi)# ip rip summary-address 10.200.200.0/24
```

ipv6 rip metric

This command sets the metric value on the interface.

The use of a negative form (no) of the command sets the default metric value.

Syntax

```
ipv6 rip metric <VALUE>
no ipv6 rip metric
```

Parameters

<VALUE> – metric value, defines in values [1..15].

Default value

5

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 rip metric 11
```

ipv6 rip summary-address

This command enables IPv6 subnetting.

The use of a negative form (no) of the command disables subnetting.

Syntax

```
[no] ipv6 rip summary-address <ADDR/LEN>
```

Parameters

<ADDR/LEN> – IP subnet, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32].

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 rip summary-address fb00::1/20
```

ipv6 router rip

The command performs the switch to RIPNG process parameters configuration mode.

The use of a negative form (no) of the command sets the default RIPNG process parameters values.

Syntax

```
[no] ipv6 router rip
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 router rip
esr(config-ripng)#
```

passive-interface

This command disables route advertising by the interface.

The use of a negative form (no) of the command restores route advertising.

Syntax

```
[no] passive-interface { <IF> | <TUN> }
```

Parameters

<IF> – an interface, specified in the form described in Section Types and naming order of router interfaces.

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

Required privilege level

10

Command mode

CONFIG-RIP

Example

```
esr(config-rip)# passive-interface gigabitethernet 1/0/15
```

passive-interface

This command disables route advertising by the interface.

The use of a negative form (no) of the command restores route advertising.

Syntax

```
[no] passive-interface <IF>
```

Parameters

<IF> – an interface, specified in the form described in Section Types and naming order of router interfaces.

Required privilege level

10

Command mode

CONFIG-RIPNG

Example

```
esr(config-ripng)# passive-interface gigabitethernet 1/0/15
```

router rip

The command performs the switch to RIP process parameters configuration mode.

The use of a negative form (no) of the command sets the default RIP process parameters values.

Syntax

```
[no] router rip
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# router rip
esr(config-rip)#
```

show ip rip

This command displays the RIP routing table.

Syntax

`show ip rip`

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip rip Sword# sho ip rip
10.10.0.1/32    via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
10.1.90.0/24    via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
192.168.16.0/24 via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
```

show ipv6 rip

This command displays the RIPNG routing table.

Syntax

`show ipv6 rip`

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 rip Sword# sho ip rip
10.10.0.1/32    via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
10.1.90.0/24    via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
192.168.16.0/24 via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
```

timers flush

This command sets the time interval after which the route will be deleted. When setting the value, consider the following rule: «timers invalid + 60».

The use of a negative form (no) of the command sets the default value.

Syntax

```
timers flush <TIME>
no timers flush
```

Parameters

<TIME> – time in seconds, takes values of [12..65535].

Default value

240 seconds

Required privilege level

10

Command mode

CONFIG-RIP

CONFIG-RIPNG

Example

```
esr(config-rip)# timers flush 300
```

timers invalid

This command defines the time interval of the route entry correctness without updating. Upon expiration, without receiving the update, the route is marked as not available.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timers invalid <TIME>
no timers invalid
```

Parameters

<TIME> – time in seconds, takes values of [12..65535].

Default value

180

Required privilege level

10

Command mode

CONFIG-RIP

CONFIG-RIPNG

Example

```
esr(config-rip)# timers invalid 240
```

timers update

This command sets the time interval after which the advertising is provided

The use of a negative form (no) of the command sets the default value.

Syntax

```
timers update <TIME>
no timers update
```

Parameters

<TIME> – time in seconds, takes values of [12..65535].

Default value

180

Required privilege level

10

Command mode

CONFIG-RIP

CONFIG-RIPNG

Example

```
esr(config-rip)# timers update 25
```

OSPF and OSPFv3 configuration

- area
- area-type
- authentication key chain
- clear ip ospf
- clear ipv6 ospf
- compatible rfc1583
- dead-interval
- hello-interval
- ip ospf
- ip ospf area
- ip ospf authentication algorithm
- ip ospf authentication key
- ip ospf authentication key-chain
- ip ospf cost
- ip ospf dead-interval
- ip ospf hello-interval
- ip ospf instance
- ip ospf mtu-ignore
- ip ospf neighbor
- ip ospf network
- ip ospf poll-interval
- ip ospf priority
- ip ospf retransmit-interval
- ip ospf wait-interval
- ipv6 ospf
- ipv6 ospf area
- ipv6 ospf cost
- ipv6 ospf dead-interval
- ipv6 ospf hello-interval
- ipv6 ospf instance
- ipv6 ospf mtu-ignore
- ipv6 ospf neighbor
- ipv6 ospf network
- ipv6 ospf poll-interval
- ipv6 ospf priority
- ipv6 ospf retransmit-interval
- ipv6 ospf wait-interval
- ipv6 router ospf
- ipv6 router ospf log-adjacency-changes
- preference
- retransmit-interval
- router ospf
- router ospf log-adjacency-changes
- router-id
- show ip ospf
- show ip ospf database
- show ip ospf interface
- show ip ospf neighbors
- show ip ospf virtual-links
- show ipv6 ospf
- show ipv6 ospf database

- show ipv6 ospf interface
- show ipv6 ospf neighbors
- show ipv6 ospf virtual-links
- summary-address
- virtual-link
- wait-interval

area

This command sets area identifier.

The use of a negative form (no) of the command removes created area.

Syntax

```
[no] area <AREA_ID>
```

Parameters

<AREA_ID> – area identifier, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-OSPF

CONFIG-OSPFV3

Example 1

```
esr(config-ospf)# area 11.11.11.51
```

Example 2

```
esr(config-ipv6-ospf)# area 11.11.11.51
```

area-type

This command defines area type.

The use of a negative form (no) of the command sets area type as default.

Syntax

```
[no] area-type <TYPE> [ no-summary ]
```

Parameters

<TYPE> – area type:

- stub – sets stub value (stub area);

no-summary – the command in conjunction with the 'stub' parameter forms the 'totally stubby' area (only the default route is used to transfer information outside the area).

- nssa – sets nssa value (NSSA area);

no-summary – the command in conjunction with the 'nssa' parameter forms the 'totally nssa' area (automatically generates the default route as interareal).

Required privilege level

10

Command mode

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

Example 1

```
esr(config-ospf-area)# area-type stub
```

Example 2

```
esr(config-ipv6-ospf-area)# area-type nssa
```

authentication key chain

The command specifies a list of passwords for authentication via md5 hash algorithm with neighbor.

The use of a negative form (no) of the command removes a bind to the password list.

Syntax

```
authentication key chain <KEYCHAIN>
no authentication key chain
```

Parameters

<KEYCHAIN> – key list identifier, set by the string of up to 16 characters.

Required privilege level

15

Command mode

CONFIG-OSPF-VLINK

Example

```
esr(config-ospf-vlink)# authentication key chain key2
```

clear ip ospf

This command resets all or a specific OSPF process.

Syntax

```
clear ip ospf [ <ID> ] [ vrf <VRF> ]
```

Parameters

<ID> – process number, takes values of [1..65535].

<VRF> – VRF instance name, set by the string of up to 31 characters, for which all or one specified OSPF process will be reset.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear ip ospf
esr# clear ip ospf 1000
```

clear ipv6 ospf

This command resets all or a specific OSPFv3 process.

Syntax

```
clear ipv6 ospf [ <ID> ] [ vrf <VRF> ]
```

Parameters

<ID> – process number, takes values of [1..65535];

<VRF> – VRF instance name, set by the string of up to 31 characters, for which all or one specified OSPFv3 process will be reset.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear ipv6 ospf
esr# clear ipv6 ospf 1000
```

compatible rfc1583

This command enables RFC 1583 compatibility.

The use of a negative form (no) of the command disables RFC 1583 compatibility.

Syntax

```
[no] compatible rfc1583
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-OSPF

CONFIG-OSPFV3

Example

```
esr(config-ospf)# compatible rfc1583
```

dead-interval

The command specifies the time interval in seconds after which the neighbor will be considered unavailable. This interval should be a multiple of the ‘hello interval’ value. As a rule, the 'dead-interval' is equal to 4 intervals of sending hello-packets, that is, 40 seconds.

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
dead-interval <TIME>
no dead-interval
```

Parameters

<TIME> – time in seconds, takes values of [1..65535].

Default value

40

Required privilege level

10

Command mode

CONFIG-OSPF-VLINK

CONFIG-OSPFV3-VLINK

Example

```
esr(config-ospf-vlink)# dead-interval 60
```

hello-interval

The command specifies the time interval in seconds after which the router sends next hello packet.

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
hello-interval <TIME>
no hello-interval
```

Parameters

<TIME> – time in seconds, takes values of [1..65535].

Default value

10

Required privilege level

10

Command mode

CONFIG-OSPF-VLINK

CONFIG-OSPFV3-VLINK

Example

```
esr(config-ospf-vlink)# hello-interval 8
```

ip ospf

This command enables routing by OSPF protocol on the interface.

The use of a negative form (no) of the command disables routing by OSPF protocol on the interface.

Syntax

`[no] ip ospf`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf
```

ip ospf area

This command adds a bind of interface to a specified OSPF process area.

The use of a negative form (no) of the command removes a bind of interface to a specified OSPF process area.

Syntax

```
ip ospf area <AREA_ID>
```

```
no ip ospf area
```

Parameters

<AREA_ID> – area identifier, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-ip4ip4)# ip ospf area 1.1.1.1
```

ip ospf authentication algorithm

This command defines authentication algorithm.

The use of a negative form (no) of the command disables the authentication.

Syntax

```
ip ospf authentication algorithm <ALGORITHM>
no ip ospf authentication algorithm
```

Parameters

<ALGORITHM> – authentication algorithm:

- cleartext – password, transmitted in clear text;
- md5 – password is hashed by md5 algorithm.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf authentication algorithm cleartext
```

ip ospf authentication key

This command sets a password for authentication with a neighbor when sending a password in clear text.

The use of a negative form (no) of the command removes the password.

Syntax

```
ip ospf authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no ip ospf authentication key
```

Parameters

<CLEAR-TEXT> – password, set by the string of 8 characters;

<ENCRYPTED-TEXT> – encrypted password of 8 bytes (16 characters) in hexadecimal format (0xYYYY ...) or (YYYY ...).

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf authentication key ascii-text 123456789
esr(config-if-gi)# ip ospf authentication key ascii-text encrypted CDE65039E5591FA3F1
```

ip ospf authentication key-chain

The command specifies a list of passwords for authentication via md5 hash algorithm with neighbor.

The use of a negative form (no) of the command removes a bind to the password list.

Syntax

```
ip ospf authentication key-chain <KEYCHAIN>
```

```
no ip ospf authentication key-chain
```

Parameters

<KEYCHAIN> – key list identifier, set by the string of up to 16 characters.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf authentication key-chain lock
```

ip ospf cost

This command sets the metric value on the interface or tunnel.

The use of a negative form (no) of the command sets the default metric value.

Syntax

```
ip ospf cost <VALUE>
no ip ospf cost
```

Parameters

<VALUE> – metric size, takes values of [0..32767].

Default value

150

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf cost 11
```

ip ospf dead-interval

The command specifies the time interval in seconds after which the neighbor will be considered unavailable. This interval should be a multiple of the «hello interval» value. As a rule, the dead-interval is equal to 4 intervals of sending hello-packets, that is, 40 seconds.

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
ip ospf dead-interval <TIME>
no ip ospf dead-interval
```

Parameters

<TIME> – time in seconds, takes values of [1..65535].

Default value

40

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf dead-interval 60
```

ip ospf hello-interval

The command specifies the time interval in seconds after which the router sends next hello packet.

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
ip ospf hello-interval <TIME>
no ip ospf hello-interval
```

Parameters

<TIME> – time in seconds, takes values of [1..255].

Default value

10

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf hello-interval 8
```

ip ospf instance

This command adds a bind of interface to a specified OSPF process.

The use of a negative form (no) of the command removes a bind of interface to a specified OSPF process.

Syntax

```
ip ospf instance <ID>
no ip ospf instance
```

Parameters

<ID> – process number, takes values of [1..65535].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-ip4ip4)# ip ospf instance 300
```

ip ospf mtu-ignore

This command enables the mode in which the OSPF process will ignore the MTU interface value in incoming Database Description packets.

The use of a negative form (no) of the command disables MTU interface ignorance mode.

Syntax

```
[no] ip ospf mtu-ignore
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf mtu-ignore
```

ip ospf neighbor

This command statically sets the IP address of the neighbor to establish a relationship in NMBA and P2MP (Point-to-MultiPoint) networks. The use of a negative form (no) of the command removes a statically specified neighbor address.

Syntax

```
[no] ip ospf neighbor <IP> [ eligible ]
```

Parameters

<IP> – neighbor IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

eligible – optional parameter, allows the device to take part in DR selection process in NMBA networks. The priority of the interface must be greater than zero, the command for changing the priority is described in section ip ospf priority.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf neighbor 10.0.0.2
```

ip ospf network

This command defines network type.

The use of a negative form (no) of the command sets the default value.

Syntax

`ip ospf network <TYPE>`

`no ip ospf network`

Parameters

<TYPE> – network type:

- broadcast – broadcast connection type;
- non-broadcast – NBMA connection type;
- point-to-multipoint – point-to-multipoint connection type;
- point-to-multipoint non-broadcast – point-to-multipoint NBMA connection type;
- point-to-point – point-to-point connection type.

Default value

broadcast

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf network point-to-point
```

ip ospf poll-interval

This command sets the time interval during which the NBMA interface waits before sending a HELLO packet to the neighbor, even if the neighbor is inactive.

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
ip ospf poll-interval <TIME>
no ip ospf poll-interval
```

Parameters

<TIME> – time in seconds, takes values of [1..255].

Default value

120

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf poll-interval 60
```

ip ospf priority

This command sets the priority of the router, which is used to select DR and BDR.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip ospf priority <VALUE>
no ip ospf priority
```

Parameters

<VALUE> – interface priority, takes values of [0..255].

Default value

120

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf priority 200
```

ip ospf retransmit-interval

This command sets the time interval in seconds after which the router will re-send a packet to which it has not received receiption confirmation (for example, Database Description packet or Link State Request packets).

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
ip ospf retransmit-interval <TIME>
no ip ospf retransmit-interval
```

Parameters

<TIME> – time in seconds, takes values of [2..65535].

Default value

5

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)#ip ospf restransmit-interval 4
```

ip ospf wait-interval

The command defines the time interval in seconds after which the router selects DR in the network.

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
ip ospf wait-interval <TIME>
no ip ospf wait-interval
```

Parameters

<TIME> – time in seconds, takes values of [1..65535].

Default value

40

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ip ospf  wait-interval 60
```

ipv6 ospf

This command enables routing by OSPFv3 protocol on the interface.

The use of a negative form (no) of the command disables routing by OSPFv3 protocol on the interface.

Syntax

```
[no] ipv6 ospf
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 ospf
```

ipv6 ospf area

This command adds a bind of interface to a specified OSPFv3 process area.

The use of a negative form (no) of the command removes a bind of interface to a specified OSPFv3 process area.

Syntax

```
ipv6 ospf area <AREA_ID>
no ipv6 ospf area
```

Parameters

<AREA_ID> – area identifier, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-ip4ip4)# ipv6 ospf area 1.1.1.1
```

ipv6 ospf cost

This command sets the metric value on the interface or tunnel.

The use of a negative form (no) of the command sets the default metric value.

Syntax

```
ipv6 ospf cost <VALUE>
no ipv6 ospf cost
```

Parameters

<VALUE> – metric value, defines in values [1..65535].

Default value

150

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 ospf cost 11
```

ipv6 ospf dead-interval

The command specifies the time interval in seconds after which the neighbor will be considered unavailable. This interval should be a multiple of the «hello interval» value. As a rule, the dead-interval is equal to 4 intervals of sending hello-packets, that is, 40 seconds.

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
ipv6 ospf dead-interval <TIME>
```

```
no ipv6 ospf dead-interval
```

Parameters

<TIME> – time in seconds, takes values of [1..65535].

Default value

40

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 ospf dead-interval 60
```

ipv6 ospf hello-interval

The command specifies the time interval in seconds after which the router sends next hello packet.

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
ipv6 ospf hello-interval <TIME>
no ipv6 ospf hello-interval
```

Parameters

<TIME> – time in seconds, takes values of [1..255].

Default value

10

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 ospf hello-interval 8
```

ipv6 ospf instance

This command adds a bind of interface to a specified OSPFv3 process.

The use of a negative form (no) of the command removes a bind of interface to a specified OSPFv3 process.

Syntax

```
ipv6 ospf instance <ID>
no ipv6 ospf instance
```

Parameters

<ID> – process number, takes values of [1..65535].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-ip4ip4)# ipv6 ospf instance 300
```

ipv6 ospf mtu-ignore

This command enables the mode in which the OSPFv3 process will ignore the MTU interface value in incoming Database Description packets.

The use of a negative form (no) of the command disables MTU interface ignorance mode.

Syntax

```
[no] ipv6 ospf mtu-ignore
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 ospf mtu-ignore
```

ipv6 ospf neighbor

This command statically sets the IPv6 address of the neighbor to establish a relationship in NMBA and P2MP (Point-to-MultiPoint) networks. The use of a negative form (no) of the command removes a statically specified neighbor address.

Syntax

```
[no] ipv6 ospf neighbor <IPV6-ADDR> [ eligible ]
```

Parameters

<IPV6-ADDR> – neighbor's IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

eligible – optional parameter, allows the device to take part in DR selection process in NMBA networks. The priority of the interface must be greater than zero, the command for changing the priority is described in section ip ospf priority.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 ospf neighbor fc00::2
```

ipv6 ospf network

This command defines network type.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 ospf network <TYPE>
no ipv6 ospf network
```

<TYPE> – network type:

- broadcast – broadcast connection type;
- non-broadcast – NBMA connection type;
- point-to-multipoint – point-to-multipoint connection type;
- point-to-multipoint non-broadcast – point-to-multipoint NBMA connection type;
- point-to-point – point-to-point connection type.

Default value

broadcast

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 ospf network point-to-point
```

ipv6 ospf poll-interval

This command sets the time interval during which the NBMA interface waits before sending a HELLO packet to the neighbor, even if the neighbor is inactive.

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
ipv6 ospf poll-interval <TIME>
no ipv6 ospf poll-interval
```

Parameters

<TIME> – time in seconds, takes values of [1..255].

Default value

120 seconds

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 ospf poll-interval 60
```

ipv6 ospf priority

This command sets the priority of the router, which is used to select DR and BDR.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 ospf priority <VALUE>
no ipv6 ospf priority
```

Parameters

<VALUE> – interface priority, takes values of [0..255].

Default value

120

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 ospf priority 300
```

ipv6 ospf retransmit-interval

This command sets the time interval in seconds after which the router will re-send a packet to which it has not received receiption confirmation (for example, Database Description packet or Link State Request packets).

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
ipv6 ospf retransmit-interval <TIME>
no ipv6 ospf retransmit-interval
```

Parameters

<TIME> – time in seconds, takes values of [2..65535].

Default value

5

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)#ipv6 ospf restransmit-interval 4
```

ipv6 ospf wait-interval

The command defines the time interval in seconds after which the router selects DR in the network.

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
ipv6 ospf wait-interval <TIME>
no ipv6 ospf wait-interval
```

Parameters

<TIME> – time in seconds, takes values of [1..65535].

Default value

40 seconds

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 ospf  wait-interval 60
```

ipv6 router ospf

This command adds OSPFv3 process to the system and switchs to the OSPFv3 process parameters configuration mode.

The use of a negative form (no) of the command removes OSPFv3 process from the system.

Syntax

```
[no] ipv6 router ospf <ID> [vrf <VRF>]
```

Parameters

<ID> – autonomous system number, takes values of [1..65535].

<VRF> – VRF instance name for OSPFv3 process, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 router ospf 300
esr(config-ipv6-ospf)#
```

ipv6 router ospf log-adjacency-changes

This command enables the display of information about the status of relations with neighbors for the OSPFv3 routing protocol.

The use of a negative form (no) of the command disables information display.

Syntax

`[no] ipv6 router ospf log-adjacency-changes`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 router ospf log-adjacency-changes
```

preference

This command defines OSPF/OSPFv3 process routes priority.

The use of a negative form (no) of the command sets the default value.

Syntax

`preference <VALUE>`

`no preference`

Parameters

<VALUE> – OSPF process routes precedence, takes values in the range of [1..255].

Default value

10

Required privilege level

10

Command mode

CONFIG-OSPF

CONFIG-OSPFV3

Example 1

```
esr(config-ospf)# preference 30
```

Example 2

```
esr(config-ipv6-ospf)# preference 30
```

retransmit-interval

This command sets the time interval in seconds after which the router will re-send a packet to which it has not received receiption confirmation (for example, Database Description packet or Link State Request packets).

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
retransmit-interval <TIME>
no retransmit-interval
```

Parameters

<TIME> – time in seconds, takes values of [2..65535].

Default value

5

Required privilege level

10

Command mode

CONFIG-OSPF-VLINK

CONFIG-OSPFV3-VLINK

Example

```
esr(config-ospf-vlink)# restransmit-interval 4
```

router ospf

This command adds OSPF process to the system and switchs to the OSPF process parameters configuration mode.

The use of a negative form (no) of the command removes a OSPF process from the system.

Syntax

```
[no] router ospf <ID> [vrf <VRF>]
```

Parameters

<ID> – stand alone system number, takes values of [1..65535];

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the routing protocol will operate.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# router ospf 300
esr(config-ospf)#
```

router ospf log-adjacency-changes

This command enables the display of information about the status of relations with neighbors for the OSPFv2 routing protocol.

The use of a negative form (no) of the command disables information display.

Syntax

```
[no] router ospf log-adjacency-changes
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# router bgp log-adjacency-changes
```

router-id

This command sets router identifier.

The use of a negative form (no) of the command removes the identifier.

Syntax

```
router-id <ID>
no router-id
```

Parameters

<ID> – router identifier, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-OSPF

CONFIG-OSPFV3

Example

```
esr(config-ospf)# router-id 1.1.1.1
```

show ip ospf

This command displays the OSPF routing table if no argument is specified. When specifying a process, displays the interface configuration information for the process.

Syntax

```
show ip ospf [ <ID> ] [ vrf <VRF> ]
```

Parameters

<ID> – process number, takes values of [1..65535];

<VRF> – VRF instance name, set by the string of up to 31 characters, for which the OSPF routing table will be displayed.

Required privilege level

1

Command mode

ROOT

Example

Routing table displaying.

```
esr# show ip ospf
O       2.2.2.0/24         [150/10] dev gi1/0/1                        [ospf2 19:40:31]  (2.2.
2.2)
```

show ip ospf database

This command displays the OSPF data table.

Syntax

```
show ip ospf <ID> [ vrf <VRF> ] database
```

Parameters

<ID> – OSPF process number, takes values of [1..65535];

<VRF> – VRF instance name for OSPF process, set by the string of up to 31 characters, within which the data table will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip ospf 111 dababase
Global
 Type   LS ID           Router           Age  Sequence  Checksum
 0005  10.166.11.12    10.1.0.1         1020  80000013    01b7
 0005  0.0.0.0         10.166.11.1       245  80000010    aa48
 0005  10.62.19.128    10.166.11.1       725  8000000e    6d2b
 0005  10.62.20.0      10.166.11.1       731  8000000d    69af
 0005  10.62.20.128    10.166.11.1       244  80000010    5e37
 0005  10.62.21.128    10.166.11.1       244  80000010    5341
 0005  10.166.11.0     10.166.11.1       245  80000010    cc6d
 0005  10.166.11.12    10.166.11.1       245  80000010    54d9
Area 0.0.11.1
 Type   LS ID           Router           Age  Sequence  Checksum
 0001  10.1.0.1        10.1.0.1         1015  80000067    989e
 0001  10.166.11.1     10.166.11.1      1021  80000018    8d96
 0002  10.166.11.14    10.166.11.1      1021  80000001    68a5
```

show ip ospf interface

This command displays OSPF interface information.

Syntax

```
show ip ospf interface [  vrf <VRF> ] [ <IF> | <TUN> ]
```

Parameters

<IF> – an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces.

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

<VRF> – VRF instance name for OSPF process, set by the string of up to 31 characters, within which the information about OSPF interface will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip ospf interface gi1/0/1
Interface:                     gigabitethernet 1/0/1
Internet Address:              25.25.0.1/24
Router ID:                     6.0.0.1
Network Type:                  broadcast
Area:                          0.0.0.0 (0)
Interface has:                 no authentication
Transmit:                      1
State:                         dr
Priority:                      128
Cost:                          10
ECMP weight:                   1
Hello timer:                   10
Wait timer:                    40
Dead timer:                    40
Retransmit timer:              5
Designed router (ID):          6.0.0.1
Designed router (IP):          25.25.0.1
Backup designed router (ID):   6.0.0.3
Backup designed router (IP):   25.25.0.3
Neighbor Count:                0
Adjacent neighbor count:       0
```

show ip ospf neighbors

This command displays information about all the neighbors or neighbors of a specific OSPF process.

Syntax

```
show ip ospf [ <ID> [ vrf <VRF> ] ] neighbors
show ip ospf neighbors[ <ID> [ vrf <VRF> ] ]
```

Parameters

<ID> – OSPF process number, takes values of [1..65535];

<VRF> – VRF instance name for OSPF process, set by the string of up to 31 characters, within which the information about neighbors will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip ospf neighbors
Router ID       Pri      State      DTime   Interface  Router IP
160.0.0.2         0     full/ptp    00:53   vlink0     160.0.0.2
95.0.0.1          1     full/dr     00:31   gi1_15     115.0.0.10
10.100.100.2    128     full/ptp    00:37   gre_25     25.25.0.2
153.0.0.1         1     full/bdr    00:30   po1        1.1.0.2
10.100.100.2    128     2way/other  00:34   gi1_14.25  14.25.0.2
24.24.24.24      15     full/bdr    00:32   te1_1      24.0.0.2
```

show ip ospf virtual-links

This command displays information about virtual connections.

Syntax

```
show ip ospf <ID> [ vrf <VRF> ] virtual-links
```

Parameters

<ID> – OSPF process number, takes values of [1..65535];

<VRF> – VRF instance name for OSPF process, set by the string of up to 31 characters, within which the information about virtual connections will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip ospf 10 virtual-links
Virtual Link to router 160.0.0.2 is ptp
Peer IP: 160.0.0.2
Transit area: 1.1.1.1
Interface has no authentication
Timer intervals configured Hello 10, Dead 60, Retransmit 5, Wait 60
Adjacency State full
```

show ipv6 ospf

This command displays the OSPFv3 routing table if no argument is specified. When specifying a process, displays the interface configuration information for the process.

Syntax

`show ipv6 ospf [ <ID> ] [ vrf <VRF> ]`

Parameters

<ID> – process number, takes values of [1..65535].

<VRF> – VRF instance name, set by the string of up to 31 characters, for which the OSPFv3 routing table will be displayed.

Required privilege level

1

Command mode

ROOT

Example

Routing table displaying.

```
esr# show ipv6 ospf
O       fc00::/120         [150/10] dev gi1/0/5               [ospf2 19:39:18]  (2.2.2.2)
```

show ipv6 ospf database

This command displays the OSPFv3 data table.

Syntax

`show ipv6 ospf <ID> [vrf <VRF>] database`

Parameters

<ID> – process number, takes values of [1..65535];

<VRF> – VRF instance name for OSPFv3 process, set by the string of up to 31 characters, within which the data table will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 ospf 111 dababase
Global
 Type   LS ID            Router           Age  Sequence  Checksum
 0005  10.166.11.12    10.1.0.1        1020  80000013    01b7
 0005  0.0.0.0         10.166.11.1      245  80000010    aa48
 0005  10.62.19.128    10.166.11.1      725  8000000e    6d2b
 0005  10.62.20.0      10.166.11.1      731  8000000d    69af
 0005  10.62.20.128    10.166.11.1      244  80000010    5e37
 0005  10.62.21.128    10.166.11.1      244  80000010    5341
 0005  10.166.11.0     10.166.11.1      245  80000010    cc6d
 0005  10.166.11.12    10.166.11.1      245  80000010    54d9
Area 0.0.11.1
 Type   LS ID            Router           Age  Sequence  Checksum
 0001  10.1.0.1        10.1.0.1        1015  80000067    989e
 0001  10.166.11.1     10.166.11.1     1021  80000018    8d96
 0002  10.166.11.14    10.166.11.1     1021  80000001    68a5
```

show ipv6 ospf interface

This command displays information about the interfaces on which OSPFv3 protocol is enabled.

Syntax

```
show ipv6 ospf interface [ vrf <VRF> ] [ <IF> ]
```

Parameters

<ID> – process number, takes values of [1..65535];

<VRF> – VRF instance name for OSPFv3 process, set by the string of up to 31 characters, within which the information about OSPFv3 interface will be displayed.

<IF> – an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 ospf interface gigabitethernet 1/0/14
esr# Interface:                    gigabitethernet 1/0/14
Internet Address:             IID 0
Router ID:                    88.88.88.88
Network Type:                 broadcast
Area:                         0.0.0.0 (0)
Transmit:                     1
State:                        backup
Priority:                     128
Cost:                         10
ECMP weight:                  1
Hello timer:                  10
Wait timer:                   40
Dead timer:                   40
Retransmit timer:             5
Designed router (ID):         77.0.0.1
Designed router (IP):         fe80::c602:46ff:feed:0
Backup designed router (ID):  88.88.88.88
Backup designed router (IP):  fe80::1:2ff:fe03:463
Neighbor Count:               1
Adjacent neighbor count:      1
Adjacent with neighbor:    77.0.0.1 (dr)
```

show ipv6 ospf neighbors

This command displays information about all the neighbors or neighbors of a specific OSPFv3 process.

Syntax

```
show ipv6 ospf [<ID> [vrf <VRF>]] neighbors
```

Parameters

<ID> – process number, takes values of [1..65535], optional parameter.

<VRF> – VRF instance name for OSPFv3 process, set by the string of up to 31 characters, within which the information about neighbors will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 ospf neighbor
Router ID          Pri      State      DTime   Interface   Router IP
77.0.0.1             1     full/dr     00:32   gi1_14      fe80::c602:46ff:feed:0
33.33.33.33        128     full/bdr    00:35   gi1_18      fe80::20:3ff:fea0:498
```

show ipv6 ospf virtual-links

This command displays information about virtual connections.

Syntax

```
show ipv6 ospf <ID> [vrf <VRF>] virtual-links
```

Parameters

<ID> – process number, takes values of [1..65535];

<VRF> – VRF instance name for OSPFv3 process, set by the string of up to 31 characters, within which the information about virtual connections will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 ospf 10 virtual-links
Virtual Link to router 160.0.0.2 is ptp
Peer IP: fe80::20:3ff:fea0:498
Transit area: 1.1.1.1
Interface has no authentication
Timer intervals configured: Hello 10, Dead 60, Retransmit 5, Wait 60
Adjacency State full
```

summary-address

This command enables subnets summation or hiding.

The use of a negative form (no) of the command disables subnets summation or hiding.

Syntax

```
summary-address { <ADDR/LEN> | <IPV6-ADDR/LEN> } { advertise | not-advertise }
no summary-address <ADDR/LEN>
```

Parameters

<ADDR/LEN> – IP subnet, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32];

<IPV6-ADDR/LEN> – IPv6 address and mask of a subnet, defined as X:X:X:X::X/EE where each X part takes values in hexadecimal format [0..FFFF] and EE takes values of [1..128];

advertise – if a command is specified, instead of the specified subnets, the total subnet will be advertised;

not-advertise – if you specify a subnet command, subnets included in the specified subnet will not be advertised.

Required privilege level

10

Command mode

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

Example

```
esr(config-ospf-area)# summary-address 192.168.16.0/24
```

virtual-link

This command establishes a virtual connection between the main and remote areas, that have several areas in between.

The use of a negative form (no) of the command removes a specified virtual connection.

Syntax

```
[no] virtual-link <ID>
```

Parameters

<ID> – router identifier with which virtual connection is establishing, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

Example

```
esr(config-ospf-area)# virtual-link 160.0.0.2
```

wait-interval

The command defines the time interval in seconds after which the router selects DR in the network.

The use of a negative form (no) of the command sets the default time interval value.

Syntax

`wait-interval <TIME>`

`no wait-interval`

Parameters

<TIME> – time in seconds, takes values of [1..65535].

Default value

40

Required privilege level

10

Command mode

CONFIG-OSPF-VLINK

CONFIG-OSPFV3-VLINK

Example

```
esr(config-ospf-vlink)# ospf wait-interval 60
```

IS-IS protocol configuration

- address-family
- authentication area algorithm
- authentication area key
- authentication area key chain
- authentication domain algorithm
- authentication domain key
- authentication domain key chain
- clear isis
- hostname dynamic
- is-type
- isis adjacency point-to-point mode
- isis authentication algorithm
- isis authentication key
- isis authentication key chain
- isis circuit-type
- isis csnp-interval
- isis enable
- isis hello-interval
- isis hello-multiplier
- isis instance
- isis lsp-interval
- isis lsp-retransmit-interval
- isis metric
- isis network point-to-point
- isis priority
- isis psnp-interval
- lsp-refresh-interval
- max-lsp-lifetime
- metric-style
- net
- no isis hello-padding
- preference
- router isis
- show ip isis
- show ipv6 isis
- show isis database
- show isis hostname
- show isis neighbors
- show isis topology
- spf-timeout

address-family

This command enables the exchange of routes of the specified family over the ISIS protocol.
The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
[no] address-family { ipv4 | ipv6 }
```

Parameters

- ipv4 – ipv4 family;
- ipv6 – ipv6 family.

Default value

ipv4 – enabled;

ipv6 – disabled.

Required privilege level

10

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# address-family ipv4
```

authentication area algorithm

This command allows to select the authentication algorithm for the L1 layer, LSP/SNP packets will be encrypted.

The use of a negative form (no) of the command sets the default time interval value.

Syntax

```
authentication area algorithm <ALGORITHM>
no authentication area algorithm
```

Parameters

<ALGORITHM> – authentication algorithm:

- cleartext – password, transmitted in clear text;
- md5 – password is hashed by md5 algorithm.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# authentication area algorithm cleartext
```

authentication area key

This command sets the authentication password for the L1 layer, this password will encrypt LSP/SNP packets.

The use of a negative form (no) of the command removes the password.

Syntax

```
authentication area key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no authentication area key
```

Parameters

<CLEAR-TEXT> – password, sets by string of [8..16] characters;

<ENCRYPTED-TEXT> – encrypted password of [8..16] bytes ([16..32] characters) in hexadecimal format (0xYYYY...) or (YYYY...).

Required privilege level

15

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# authentication area key ascii-text password
```

authentication area key chain

This command sets the list of keys for L1 level authentication.

The use of a negative form (no) of the command removes a bind to the key list.

Syntax

```
authentication area key chain <KEYCHAIN>
no authentication area key chain
```

Parameters

<KEYCHAIN> – key list identifier, set by the string of [1..16] characters.

Required privilege level

15

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# authentication area key chain key-1
```

authentication domain algorithm

This command sets the authentication algorithm for the L2 layer, LSP/SNP packets will be encrypted.

The use of a negative form (no) of the command disables the authentication.

Syntax

```
authentication domain algorithm <ALGORITHM>
no authentication domain algorithm
```

Parameters

<ALGORITHM> – authentication algorithm:

- cleartext – password, transmitted in clear text;
- md5 – password is hashed by md5 algorithm.

Required privilege level

15

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# authentication domain algorithm cleartext
```

authentication domain key

This command sets the authentication password for the L2 layer, this password will encrypt LSP/SNP packets.

The use of a negative form (no) of the command removes the password.

Syntax

```
authentication domain key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no authentication domain key
```

Parameters

<CLEAR-TEXT> – password, sets by string of [8..16] characters;

<ENCRYPTED-TEXT> – encrypted password of [8..16] bytes ([16..32] characters) in hexadecimal format (0xYYYY...) or (YYYY...).

Required privilege level

15

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# authentication domain key ascii-text password
```

authentication domain key chain

This command sets the list of keys for L2 level authentication.

The use of a negative form (no) of the command removes a bind to the key list.

Syntax

```
authentication domain key chain <KEYCHAIN>
no authentication domain key chain
```

Parameters

<KEYCHAIN> – key list identifier, set by the string of [1..16] characters.

Required privilege level

15

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# authentication domain key chain key-1
```

clear isis

This command restarts a specific IS-IS process. The command without parameters restarts all IS-IS processes configured on the router.

Syntax

```
clear isis [ <ID> ] [ vrf <VRF_NAME> ]
```

Parameters

<ID> – process number, takes values of [1..65535];

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear isis 15
```

hostname dynamic

This command enables the use of TLV 137, which allows the hostname of the router to be passed to the LSP.

The use of a negative form (no) of the command disables this feature.

Syntax

```
[no] hostname dynamic
```

Parameters

The command does not contain parameters

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# hostname dynamic
```

is-type

This command defines which routing layer the current IS-IS process will run on.

The use of a negative form (no) of the command sets the default value.

Syntax

```
is-type <LEVEL>
no is-type
```

Parameters

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-1-2 – operate on levels 2 and 2;
- level-2-only – operate only on level 2.

Default value

level-1-2

Required privilege level

10

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# is-type level-1
```

isis adjacency point-to-point mode

This command sets the method of establishing a point-to-point neighborhood.

The use of a negative form (no) of the command sets the default value.

Syntax

```
isis adjacency point-to-point mode <MODE>
no isis adjacency point-to-point mode
```

Parameters

<MODE> – neighborhood establishment method:

- two-way-only – IIH with TLV 240 "Point-to-Point Three-Way Adjacency" will be processed, but TLV 240 will be ignored;
- three-way-only – IIH without TLV 240 will not be processed.

Default value

All methods allowed.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis adjacency point-to-point mode three-way-only
```

isis authentication algorithm

This command sets the authentication algorithm for hello (IIH) packets coming out of this interface.

The use of a negative form (no) of the command disables the authentication.

Syntax

```
isis authentication algorithm <ALGORITHM> [ <LEVEL> ]
no isis authentication algorithm [ <LEVEL> ]
```

Parameters

<ALGORITHM> – authentication algorithm:

- cleartext – password, transmitted in clear text;
- md5 – password is hashed by md5 algorithm;

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis authentication algorithm cleartext level-1
```

isis authentication key

This command sets a password to authenticate hello packets (IIH).

The use of a negative form (no) of the command removes the password.

Syntax

```
isis authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
[ <LEVEL> ]
no isis authentication key [ <LEVEL> ]
```

Parameters

<CLEAR-TEXT> – password, sets by string of [8..16] characters;

<ENCRYPTED-TEXT> – encrypted password of [8..16] bytes ([16..32] characters) in hexadecimal format (0xYYYY...) or (YYYY...).

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis authentication key ascii-text password level-1
```

isis authentication key chain

This command sets a key list to authenticate hello packets (IIH).

The use of a negative form (no) of the command removes a bind to the key list.

Syntax

```
isis authentication key chain <KEYCHAIN> [ <LEVEL> ]
no isis authentication key chain [ <LEVEL> ]
```

Parameters

<KEYCHAIN> – key list identifier, set by the string of [1..16] characters.

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis authentication key chain key-1 level-1
```

isis circuit-type

This command defines which routing layer on the interface the current IS-IS process will run on.

The use of a negative form (no) of the command sets the operating level of the IS-IS protocol on the interface according to the configuration of the IS-IS process itself.

Syntax

```
isis circuit-type <LEVEL>
no isis circuit-type
```

Parameters

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-1-2 – operate on levels 2 and 2;
- level-2-only – operate only on level 2.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis circuit-type level-1
```

isis csnp-interval

This command sets the interval for generating and sending Complete Sequence Number Protocol Data Units (CSNP PDUs).

The use of a negative form (no) of the command sets the default value.

Syntax

```
isis csnp-interval <TIME> [ <LEVEL> ]
no isis csnp-interval [ <LEVEL> ]
```

Parameters

<TIME> – time in seconds, takes values of [1..65535];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Default value

10

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis csnp-interval 33 level-1
```

isis enable

This command enables the IS-IS protocol on the interface.

The use of a negative form (no) of the command disables IS-IS process.

Syntax

```
isis enable
no isis enable
```

Parameters

The command does not contain parameters

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis enable
```

isis hello-interval

This command sets the interval for sending hello packets (IIH) from the router interface.

The use of a negative form (no) of the command sets the default value.

Syntax

```
isis hello-interval <TIME> [ <LEVEL> ]
no isis hello-interval [ <LEVEL> ]
```

Parameters

<TIME> – time in seconds, takes values of [1..65535];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Default value

10

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis hello-interval 33 level-1
```

isis hello-multiplier

This command sets the multiplier for calculating and sending Hold Time (Hold Time = hello-timer * hello-multiplier) in hello packets (IIH).

The use of a negative form (no) of the command sets the default value.

Syntax

```
isis hello-multiplier <VALUE> [ <LEVEL> ]
no isis hello-multiplier [ <LEVEL> ]
```

Parameters

<VALUE> – number, may take values [3..1000];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Default value

3

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis hello-multiplier 34 level-1
```

isis instance

This command adds a bind of interface to a specified IS-IS process.

The use of a negative form (no) of the command removes a bind of interface to a specified IS-IS process.

Syntax

```
isis instance <ID>
no isis instance
```

Parameters

<ID> – process number, takes values of [1..65535].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis instance 35
```

isis lsp-interval

This command sets the interval between successive transmissions of Link-state Packets (LSPs) on a Broadcast network.

The use of a negative form (no) of the command sets the default value.

Syntax

```
isis lsp-interval <TIME> [ <LEVEL> ]
no isis lsp-interval [ <LEVEL> ]
```

Parameters

<TIME> – time in milliseconds, takes values of [1-10000];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Default value

100

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis lsp-interval 36 level-2
```

isis lsp-retransmit-interval

This command sets the redistribution interval for all Link-state Packets (LSPs) on the PtP network.

The use of a negative form (no) of the command sets the default value.

Syntax

```
isis lsp-retransmit-interval <TIME> [ <LEVEL> ]
no isis lsp-retransmit-interval [ <LEVEL> ]
```

Parameters

<TIME> – time in seconds, takes values of [1..65535];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Default value

5

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis lsp-retransmit-interval 37 level-2
```

isis metric

This command sets a specific metric value for the interface.

The use of a negative form (no) of the command sets the default value.

Syntax

```
isis metric <VALUE> [<LEVEL>]
no isis metric [<LEVEL>]
```

Parameters

<VALUE> – number, may take values [1..16777215];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Default value

10

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis metric 38 level-1
```

isis network point-to-point

This command sets the current interface to point-to-point IS-IS protocol mode.

The use of a negative form (no) of the command stops operation in this mode.

Syntax

`[no] isis network point-to-point`

Parameters

The command does not contain parameters

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-gre)# isis network point-to-point
```

isis priority

This command sets the priority of the DIS selection.

The use of a negative form (no) of the command sets the default value.

Syntax

```
isis priority <VALUE> [ <LEVEL> ]
no isis priority [ <LEVEL> ]
```

Parameters

<VALUE> – number, may take values [0..127];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Default value

64

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis priority 39 level-2
```

isis psnp-interval

This command sets the interval for generating and sending Partial Sequence Number Protocol Data Units (PSNP PDUs).

The use of a negative form (no) of the command sets the default value.

Syntax

```
isis psnp-interval <TIME> [ <LEVEL> ]
no isis psnp-interval [ <LEVEL> ]
```

Parameters

<TIME> – time in seconds, takes values of [1..65535];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Default value

2

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis psnp-interval 40 level-1
```

lsp-refresh-interval

This command sets the intervals for updating/generating own LSP.

The use of a negative form (no) of the command sets the default value.

Syntax

```
lsp-refresh-interval { min | max } <TIME> [ <LEVEL> ]
no lsp-refresh-interval { min | max } [ <LEVEL> ]
```

Parameters

min — minimum update/generation interval;

max — maximum update/generation interval;

<TIME> – time in seconds, takes values of [1..65535];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Default value

min 30

max 900

Required privilege level

10

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# lsp-refresh-interval min 41 level-1
```

max-lsp-lifetime

This command sets the lifetime of own LSP.

The use of a negative form (no) of the command sets the default value.

Syntax

```
max-lsp-lifetime <TIME> [ <LEVEL> ]
no max-lsp-lifetime [ <LEVEL> ]
```

Parameters

<TIME> – time in seconds, takes values of [1..65535];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Default value

1200

Required privilege level

10

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# max-lsp-lifetime 42 level-2
```

metric-style

This command sets the type of metric to be used by the IS-IS process.

The use of a negative form (no) of the command sets the default value.

Syntax

```
metric-style { narrow | wide | transition } [ <LEVEL> ]
no metric-style [ <LEVEL> ]
```

Parameters

narrow — information about the reachability of networks is disseminated with TLV 128;

wide — information about the reachability of networks is disseminated with TLV 135;

transition — information about the reachability of networks is disseminated with both methods;

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2-only – operate only on level 2.

Default value

narrow

Required privilege level

10

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# metric-style wide level-1
```

net

This command sets NET (Network Entity Title) address. It represents the AFI (Authority and Format Identifier), Area ID (zone to which the router belongs), System ID (router ID) and Selector value.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] net <NET>
```

Parameters

<NET> – NET address, format: ff[.ffff.ffff.ffff.ffff.ffff.ffff].ffff.ffff.ffff.00, where f – may take values [1..F];

Required privilege level

10

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# net 49.0123.1010.1010.1010.00
```

no isis hello-padding

This command disables the TLV 8 mode in hello packets (IIH).

The use of a positive form (without «no» key) of the command sets the default value.

Syntax

```
[no] isis hello-padding
```

Parameters

The command does not contain parameters

Default value

Enabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Example

```
esr(config-if-gi)# isis hello-padding
```

preference

This command sets the route priority of this IS-IS process to the main routing table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
preference <VALUE>
no preference
```

Parameters

<VALUE> – may take values [1..255].

Default value

160

Required privilege level

10

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# preference 43
```

router isis

This command creates an IS-IS process and enters the setting mode for that process.

The use of a negative form (no) of the command removes IS-IS process from the system.

Syntax

```
[no] router isis <ID> [ vrf <VRF> ]
```

Parameters

<ID> – process number, takes values of [1..65535];

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# router isis 44
```

show ip isis

This command outputs the internal routing table

Syntax

```
show ip isis [ <ID> ] [ <ADDR/LEN> ] [ <LEVEL> ] [ vrf <VRF> ]
```

Parameters

<ID> – process number, takes values of [1..65535];

<ADDR/LEN> – address, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2 – operate only on level 2;

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip isis 45
IPv4 local Level 1 RIB for IS-IS process
Last update: 2020-01-24 06:51:02.110
Address                       Type Metric Via
3.3.3.11/32                   L1   20     2.2.2.11 on gi1/0/2

IPv4 local Level 2 RIB for IS-IS process
Last update: 2020-01-24 06:51:12.440
Address                       Type Metric Via
3.3.3.22/32                   IA   20     1.1.1.2 on gi1/0/1
```

show ipv6 isis

This command outputs the internal ipv6 routing table

Syntax

```
show ipv6 isis [ <ID> ] [ <IPV6-ADDR/LEN> ] [ <LEVEL> ] [ vrf <VRF> ]
```

Parameters

<ID> – process number, takes values of [1..65535];

<IPV6-ADDR/LEN> – IPv6 address of subnet with mask, defined as X:X:X:X::X/EE where each X part takes values in hexadecimal format [0..FFFF] and EE takes values of [1..128];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2 – operate only on level 2;

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 isis 46
IPv6 local Level 1 RIB for IS-IS process
Last update: 2020-01-24 07:08:37.761
Address                       Type Metric Via
6666::/64                     L1   20     fe80::aaf9:4bff:feab:813a on gi1/0/2

IPv6 local Level 2 RIB for IS-IS process
Last update: 2020-01-24 07:07:22.917
Address                       Type Metric Via
2001::/64                     IA   20     fe80::aaf9:4bff:feac:b4a1 on gi1/0/1
```

show isis database

This command outputs information on the local base of all LSPs received and generated (LSDB).

Syntax

```
show isis database [ <ID> ] [ <LEVEL> ] [ <LSP-ID> ] [ detailed ] [ vrf <VRF> ]
```

Parameters

<ID> – process number, takes values of [1..65535];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2 – operate only on level 2;

<LSP-ID> – ID LSP of the packet, specified as xxxx.xxxx.xxxx.yy-zz, where x – System-ID, y – Pseudonode-ID, z – fragment number. All numbers can take values [0...F];

detailed – detailed information on selected point;

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show isis database 47
IS-IS Level 1 Link State Database
LSPID                   Hostname          LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
1010.1010.1010.00-00 * esr-l1l2          0x0000000f   0x5eba        960           1/0/0
1111.1111.1111.00-00   esr-l1            0x00000010   0x4402        1071          0/0/0
1111.1111.1111.01-00   esr-l1            0x00000004   0x1a99        985           0/0/0

IS-IS Level 2 Link State Database
LSPID                   Hostname          LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
1010.1010.1010.00-00 * esr-l1l2          0x00000010   0x7776        1082          0/0/0
1222.1222.1222.01-00   esr-l2            0x00000003   0xb87a        997           0/0/0
1222.1222.1222.00-00   esr-l2            0x0000000e   0x84c4        997           0/0/0
```

show isis hostname

This command outputs the matching of System-id and router name.

Syntax

```
show isis hostname [ <ID> ] [ vrf <VRF> ]
```

Parameters

<ID> – process number, takes values of [1..65535];

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show isis hostname 48
IS-IS 12
Level  System ID         Dynamic Hostname
   1   1010.1010.1010 * esr-l1l2
   1   1111.1111.1111   esr-l1
   2   1010.1010.1010 * esr-l1l2
   2   1222.1222.1222   esr-l2
```

show isis neighbors

This command displays information on the status of IS-IS neighbors.

Syntax

```
show isis neighbors [ <ID> ] [ <LEVEL> ] [ detailed ] [ vrf <VRF> ]
```

Parameters

<ID> – process number, takes values of [1..65535];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2 – operate only on level 2;

detailed – detailed information on selected point;

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show isis neighbors 49 detailed
IS-IS Level 1 Neighbors
System ID       Hostname          Interface         State         Holdtime  SNPA
1111.1111.1111  esr-l1            gi1/0/2           Up            10        a8f9.4bab.813a

IS-IS Level 2 Neighbors
System ID       Hostname          Interface         State         Holdtime  SNPA
1222.1222.1222  esr-l2            gi1/0/1           Up            10        a8f9.4bac.b4a1
```

show isis topology

This command outputs a complete reachability table of all networks and hosts.

Syntax

```
show isis topology [ <ID> ] [ <LEVEL> ] [ ipv4 | ipv6 ] [ vrf <VRF> ]
```

Parameters

<ID> – process number, takes values of [1..65535];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2 – operate only on level 2;

ipv4 – ipv4 family;

ipv6 – ipv6 family;

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show isis topology 50 level-2 Ipv4
IS-IS path to Level 2 routers (narrow metric-style)
Last update: 2020-01-24 07:08:47.783
Vertex                        Type             Metric   Next-Hop           Interface
1.1.1.0/24                    IPv4 Internal    0        --

IS-IS path to Level 2 routers (wide metric-style)
Last update: 2020-01-24 07:08:47.783
Vertex                        Type             Metric   Next-Hop           Interface
3.3.3.22/32                   IPv4 IA          20       esr-l2             gi1/0/1
1.1.1.0/24                    IPv4             0        --
```

spf-timeout

This command sets a timeout before the next SPF calculation.

The use of a negative form (no) of the command sets the default value.

Syntax

```
spf-timeout <TIME> [ <LEVEL> ]
no spf-timeout [ <LEVEL> ]
```

Parameters

<TIME> – time in milliseconds, takes values of [1..10000];

<LEVEL> – IS-IS protocol operation level:

- level-1 – operate only on level 1;
- level-2 – operate only on level 2.

Default value

5000

Required privilege level

10

Command mode

CONFIG-ISIS

Example

```
esr(config-isis)# spf-timeout 51 level-1
```

24 MPLS configuration

- General MPLS settings
  - forwarding interface
  - l2vpn
  - ldp
  - mpls
  - system cpu load-balance mpls passenger
- LDP settings
  - address-family
  - advertise-labels
  - clear mpls ldp neighbor
  - description
  - discovery hello holdtime
  - discovery hello interval
  - discovery targeted-hello accept
  - discovery targeted-hello interval
  - discovery targeted-hello holdtime
  - egress-label-type
  - enable
  - interface
  - keepalive
  - neighbor
  - password
  - router-id
  - show mpls ldp binding
  - show mpls ldp discovery
  - show mpls forwarding-table
  - show mpls ldp neighbor
  - targeted
  - transport-address
- Настройки MPLS L2VPN
  - autodiscovery bgp
  - ignore encapsulation-mismatch
  - ignore mtu-mismatch
  - mtu
  - rd
  - route-target
  - ve id
  - ve range
  - vpn id
  - bridge-group
  - description
  - enable
  - encapsulation mpls mtu
  - encapsulation mpls status-tlv disable
  - interface
  - neighbor-address
  - p2p
  - pw
  - pw-class
  - pw-class
  - transport-mode
  - vpls

- show mpls l2vpn bindings
- show mpls l2vpn pseudowire
- show mpls l2vpn p2p
- show mpls l2vpn pw-class
- show mpls l2vpn vpls

General MPLS settings

forwarding interface

This command enables mpls packet forwarding on the interface.

The use of a negative form (no) of the command disables mpls packet forwarding on the interface.

Syntax

```
[no] forwarding interface { <IF> | <TUN> }
```

Parameters

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-MPLS

Example

```
esr(config-mpls)# forwarding interface gi 1/0/14
```

l2vpn

This command switches to the MPLS-L2 tunnels configuration context.

The use of a negative form (no) of the command removes all settings of MPLS-L2 tunnels.

Syntax

```
[no] l2vpn
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-MPLS

Example

```
esr(config-mpls)# l2vpn
```

ldp

This command switches to the LDP configuration context.

The use of a negative form (no) of the command removes all LDP settings.

Syntax

```
[no] ldp
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-MPLS

Example

```
esr(config-mpls)# ldp
```

mpls

This command switches to the MPLS functionality configuration context.

The use of a negative form (no) of the command removes all LDP functionality settings.

Syntax

[no] mpls

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# mpls
```

system cpu load-balance mpls passenger

This command enables the traffic balancing functionality between the router CPUs.

The use of a negative form (no) of the command disables traffic balancing functionality.

Syntax

```
system cpu load-balance mpls passenger [ <OPTION> ]
no system cpu load-balance mpls passenger [ <OPTION> ]
```

Parameters

<OPTION> parameter may take values:

- ip – enables searching and adding ip-src and ip-dst to the hash calculation;
- ipoe-pw-with-cw – ability to explicitly specify that a Control Word header is used.
- ipoe-pw-without-cw – ability to explicitly specify that a Control Word header is not used.

Default value

Disabled. The hash calculation does not use ip-src and ip-dst.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# system cpu load-balance mpls passenger ip
```

LDP settings

address-family

The command allows to enter the LDP address family parameter configuration mode.

The use of a negative form (no) of the command removes all parameters from the configuration.

Syntax

```
[no] address-family ipv4
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-LDP

Example

```
esr(config-ldp)# address-family ipv4
```

advertise-labels

The command allows to set the FECs for which LDP will allocate and distribute tags.

The use of a negative form (no) of the command returns the default value

Syntax

```
advertise-labels <OBJ-GROUP-NETWORK-NAME>
no advertise-labels
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – IP addresses profile name, set by the string of up to 31 characters.

Default value

Tags are highlighted and distributed to all FECs

Required privilege level

10

Command mode

CONFIG-LDP

Example

```
esr(config-ldp)# advertise-labels mpls_adv
```

clear mpls ldp neighbor

This command is used to clear the LDP neighbor table.

Syntax

```
clear mpls ldp neighbor [ <ADDR> ]
```

Parameters

<ADDR> – LDP neighbor's IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

1

Command mode

ROOT

Example

```
ESR1# clear mpls ldp neighbor 1.1.1.1
```

description

This command allows to add a description for LDP neighbor.

The use of a negative form (no) of the command removes LDP neighbor description.

Syntax

```
description <STRING>
no description
```

Parameters

<STRING> – string from 1 to 255 characters.

Required privilege level

10

Command mode

CONFIG-LDP-NEIGH

Example

```
esr(config-ldp-neig)# description ESR1
```

discovery hello holdtime

This command sets the value of the holdtime parameter. Holdtime — time interval during which at least one LDP Hello message should arrive.

The use of a negative form (no) of the command sets the default value.

Syntax

```
discovery hello holdtime <TIME>

no discovery hello holdtime
```

Parameters

<TIME> — time in the range of [3..65535] seconds.

Default value

15

Required privilege level

10

Command mode

CONFIG-LDP

CONFIG-LDP-AF-IPV4-IF

Example

```
esr(config-ldp-af-ipv4-if)# discovery hello holdtime 45
```

discovery hello interval

This command sets the value of the hello interval parameter. Hello interval — time interval between sending the ldp hello message.

The use of a negative form (no) of the command sets the default value.

Syntax

```
discovery hello interval <TIME>
no discovery hello interval
```

Parameters

<TIME> — time in the range of [1..65535] seconds.

Default value

5

Required privilege level

10

Command mode

CONFIG-LDP

CONFIG-LDP-AF-IPV4-IF

Example

```
esr(config-ldp-af-ipv4-if)# discovery hello interval 15
```

discovery targeted-hello accept

This command allows the reception of targeted-hello messages.

The use of a negative form (no) of the command denies the reception of targeted-hello messages.

Syntax

```
discovery targeted-hello accept
no targeted-hello accept
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-LDP-AF-IPV4

Example

```
esr(config-ldp-af-ipv4)# discovery targeted-hello accept
```

discovery targeted-hello interval

This command sets the value of the hello interval parameter. Hello interval — time interval between sending the ldp hello messages for targeted neighbor.

The use of a negative form (no) of the command sets the default value.

Syntax

```
discovery targeted-hello interval <TIME>
no discovery targeted-hello interval
```

Parameters

<TIME> — time in the range of [1..65535] seconds.

Default value

5

Required privilege level

10

Command mode

CONFIG-LDP

CONFIG-LDP-NEIGH

Example

```
esr(config-ldp-neig)# discovery targeted-hello interval 15
```

discovery targeted-hello holdtime

This command sets the value of the holdtime parameter for the targeted neighbor. Holdtime — time interval during which at least one LDP Hello message should arrive.

The use of a negative form (no) of the command sets the default value.

Syntax

```
discovery targeted-hello holdtime <TIME>
```

```
no discovery targeted-hello holdtime
```

Parameters

<TIME> — time in the range of [3..65535] seconds.

Default value

45

Required privilege level

10

Command mode

CONFIG-LDP

CONFIG-LDP-NEIGH

Example

```
esr(config-ldp-af-ipv4-if)# discovery hello holdtime 45
```

egress-label-type

This command sets the mode when the explicit-null (0) tag is announced for FECs whose router is an egress lsr instead of the implicit null (3) tag.

The use of a negative form (no) of the command sets the default value.

Syntax

```
egress-label-type explicit-null
```

```
no egress-label-type
```

Parameters

The command does not contain parameters.

Default value

Advertising IMPLICIT NULL (3)

Required privilege level

10

Command mode

CONFIG-LDP

Example

```
esr(config-ldp)# egress-label-type explicit-null
```

enable

This command enables the LDP process.

The use of a negative form (no) of the command disables the LDP process.

Syntax

`[no] enable`

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-LDP

Example

```
esr(config-ldp)# enable
```

interface

This command enables the reception and sending of ldp hello messages on the interface.

The use of a negative form (no) of the command disables the reception and sending of ldp hello messages on the interface.

Syntax

```
[no] interface { <IF> | <TUN> }
```

Parameters

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

Required privilege level

10

Command mode

CONFIG-LDP-AF-IPV4

Example

```
esr(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
```

keepalive

This command is used to set the waiting time for a keepalive message from a neighbor. If the timer expires, the neighbor is considered unavailable.

The use of a negative form (no) of the command sets the default value.

Syntax

```
keepalive <TIMER>
no keepalive
```

Parameters

<TIMER> – Time in seconds [3..65535].

Default value

180

Required privilege level

10

Command mode

CONFIG-LDP

CONFIG-LDP-NEIGH

Example

```
esr(config-ldp)# keepalive 120
esr(config-ldp-neig)# keepalive 160
```

neighbor

This command allows to create and switch to the LDP-targeted neighbor configuration mode.

The use of a negative form (no) of the command removes targeted neighbor from the configuration.

Syntax

`[no] neighbor <ADDR>`

Parameters

<ADDR> – neighbor’s IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

Required privilege level

10

Command mode

CONFIG-LDP

Example

```
 esr(config-ldp)# neighbor 2.2.2.2
```

password

This command sets a password for authentication with a neighbor.

The use of a negative form (no) of the command removes the password.

Syntax

```
password { <TEXT> | encrypted <ENCRYPTED-TEXT> }
no password
```

Parameters

<CLEAR-TEXT> – password, sets by string of [8..16] characters;

<ENCRYPTED-TEXT> – encrypted password of [8..16] bytes ([16..32] characters) in hexadecimal format (0xYYYY...) or (YYYY...).

Required privilege level

10

Command mode

CONFIG-LDP-NEIGH

Example

```
esr(config-ldp-neig)# password cleartextpassword
```

router-id

This command sets the router ID for LDP.

The use of a negative form (no) of the command removes the router ID for LDP.

Syntax

```
router-id <ID>
no router-id
```

Parameters

<ID> – router identifier, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-LDP

Example

```
 esr(config-ldp)# router-id 1.1.1.1
```

show mpls ldp binding

This command allows to display information on the selected tags.

Syntax

```
show mpls ldp binding [ detailed ] [ { <ADDR/LEN> | neighbor <ADDR> | [ local <LABEL-ID> ] [ remote <LABEL-ID> ] } ]
```

Parameters

<ADDR/LEN> – IP subnet, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32].

<LABEL-ID> – label, specified in the range [0-104875].

<ADDR> – LDP neighbor's IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

detailed – information is given in the detailed format.

Required privilege level

1

Command mode

ROOT

Example

```
ESR1# sh mpls ldp bindings
10.10.0.4/32:
    local label:    16
    remote label:   16        lsr: 10.10.0.1      inuse
    remote label:   149       lsr: 10.10.0.3
    remote label:   imp-null  lsr: 10.10.0.4      inuse
10.10.0.2/32:
    local label:    imp-null
    remote label:   34        lsr: 10.10.0.1
    remote label:   152       lsr: 10.10.0.3
    remote label:   142       lsr: 10.10.0.4
10.10.0.1/32:
    local label:    20
    remote label:   imp-null  lsr: 10.10.0.1      inuse
    remote label:   139       lsr: 10.10.0.3
    remote label:   127       lsr: 10.10.0.4
```

show mpls ldp discovery

This command allows to display information on the status of the LDP discovery process

Syntax

```
show mpls ldp discovery [ detailed ]
```

Parameters

detailed – information is given in the detailed format.

Required privilege level

1

Command mode

ROOT

Example

```
ESR1# show mpls ldp discovery detailed
Local LDP ID: 1.1.1.1
Discovery sources:
Interfaces:
gigabitethernet 1/0/1:
Hello interval: 5 seconds
Transport IP address: 1.1.1.1
LDP ID: 4.4.4.4
Source IP address: 10.10.10.2
Transport IP address: 4.4.4.4
Hold time: 15 seconds
Proposed hold time: 15/15 (local/peer) seconds
Targeted hellos:
1.1.1.1 -> 4.4.4.4:
Hello interval: 5 seconds
Transport IP address: 1.1.1.1
LDP ID: 4.4.4.4
Source IP address: 4.4.4.4
Transport IP address: 4.4.4.4
Hold time: 45 seconds
Proposed hold time: 45/45 (local/peer) seconds
```

show mpls forwarding-table

This command allows to output the label switching table (LFIB)

Syntax

```
show mpls forwarding-table [ { <ADDR/LEN> | label <LABLE-ID> | nexthop <ADDR> | tunnel
<ID> } ]
```

Parameters

<ADDR/LEN> – IP subnet, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32].

<LABEL-ID> – label, specified in the range [0..104875].

<ADDR> – LDP neighbor's IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<ID> — identifier, specified in the range [1..2000000].

Required privilege level

1

Command mode

ROOT

Example

```
ESR1# show mpls forwarding-table
Local     Outgoing                   Prefix                       Outgoing
Next Hop
label      label               or tunnel ID Interface
-------- -------- ------------------------------------------- ----------------
---------------------------------------
18         16                     PW   ID 100                     --                  4.4.
4.4
17       imp-null                  4.4.4.4/32                       gi1/0/1             10.10.
10.2
```

show mpls ldp neighbor

This command displays information about all LDP neighbors.

Syntax

```
show mpls ldp neighbor [ { <ADDR> | interface { <IF> | <TUN> } } ]
```

Parameters

<ADDR> – LDP neighbor’s IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

Required privilege level

1

Command mode

ROOT

Example

```
ESR1# show mpls ldp neighbor
Peer LDP ID: 4.4.4.4; Local LDP ID 1.1.1.1
State: Operational
TCP connection: 4.4.4.4:38759 - 1.1.1.1:646
Messages sent/received: 195/194
Uptime: 03:07:33
LDP discovery sources:
gigabitethernet 1/0/1
1.1.1.1 -> 4.4.4.4
```

targeted

This command enables the sending of targeted hello messages.

The use of a negative form (no) of the command disables the sending of targeted hello messages.

Syntax

```
[no] targeted
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-LDP-NEIGH

Example

```
esr(config-ldp-neig)# targeted
```

transport-address

This command specifies the source ip-address for LDP messages over TCP.

The use of a negative form (no) of the command removes the source ip-address for LDP messages over TCP.

Syntax

```
transport-address <ADDR>
no transport-address
```

Parameters

<ADDR> – defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-LDP-AF-IPV4

Example

```
esr(config-ldp-af-ipv4)# transport-address 1.1.1.1
```

Настройки MPLS L2VPN

autodiscovery bgp

The command allows to enter the BGP auto-discovery and signaling configuration context.

The use of a negative form (no) of the command removes the parameter from the configuration.

Syntax

```
[no] autodiscovery bgp
```

Parameters

None.

Required privilege level

10

Command mode

CONFIG-L2VPN-EOMPLS

Example

```
esr(config-l2vpn-vpls)# autodiscovery bgp
```

ignore encapsulation-mismatch

This command allows a pseudowire connection to be established if the encapsulation type does not match.

The use of a negative form (no) of the command enables the encapsulation type check before installing the pseudowire connection.

Syntax

```
[ no ] ignore encapsulation-mismatch
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-AUTODISCOVERY-BGP

Example

```
esr(config-autodiscovery-bgp)# ignore encapsulation-mismatch
```

ignore mtu-mismatch

This command allows a pseudowire connection to be established if the MTU value does not match.

The use of a negative form (no) of the command enables the MTU value check before installing the pseudowire connection.

Syntax

```
[no] ignore mtu-mismatch
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-AUTODISCOVERY-BGP

Example

```
esr(config-autodiscovery-bgp)# ignore mtu-mismatch
```

mtu

This command sets the MTU value for the pseudowire connection. The set value is passed to the extended community.

The use of a negative form (no) of the command sets the default MTU value.

Syntax

```
mtu <VALUE>
no mtu
```

Parameters

<VALUE> – MTU value, may take values in the range [552..10000].

Default value

1500

Required privilege level

10

Command mode

CONFIG-AUTODISCOVERY-BGP

Example

```
esr(config-autodiscovery-bgp)# mtu 1510
```

rd

This command is used to set Route Distinguisher for the given VPLS instance.

The use of a negative form (no) of the command removes the parameter from the configuration.

Syntax

```
rd <RD>
no rd
```

Parameters

<RD> – Route distinguisher value, specified in one of the following forms:

- <ASN>:<nn> — where <ASN> may take values [1..65535], nn may take values [1..65535];
- <ADDR>:<nn> — where <ADDR> specified as AAA.BBB.CCC.DDD/EE, AAA-DDD may take values [0..255], nn may take values [1..65535];
- <4ASN>:<nn> — where <4ASN> may take values [1..4294967295], nn may take values [1..65535];

Required privilege level

10

Command mode

CONFIG-AUTODISCOVERY-BGP

Example

```
esr(config-vrf)# rd 65000:10001
```

route-target

This command is used to set the route-target extended community value list.

Syntax

```
[no] route-target { import | export } <RT>
```

Parameters

import – command defines routes from the L2VPN table with value <RT>, which will be placed to the corresponding VPLS instance.

export – command sets the <RT> values with which routes from this VPLS instance will be announced in BGP.

<RT> – route-target value, specified in one of the following forms:

- <ASN>:<nn> — where <ASN> may take values [1..65535], nn may take values [1..65535];
- <ADDR>:<nn> — where <ADDR> specified as AAA.BBB.CCC.DDD/EE, AAA-DDD may take values [0..255], nn may take values [1..65535];
- <4ASN>:<nn> — where <4ASN> may take values [1..4294967295], nn may take values [1..65535];

Required privilege level

10

Command mode

CONFIG-AUTODISCOVERY-BGP

Example

```
esr(config-vrf)# route-target export 65000:10001
```

ve id

This command sets the identifier of the VPLS instance.

The use of a negative form (no) of the command removes the parameter from the configuration.

Syntax

```
ve id <ID>
no ve id
```

Parameters

<ID> – VPLS instance identifier, takes values in the range of [1..16384].

Required privilege level

10

Command mode

CONFIG-AUTODISCOVERY-BGP

Example

```
esr(config-autodiscovery-bgp)# ve id 5
```

ve range

This command configures the range of VPLS edge device IDs.

The use of a negative form (no) of the command sets the default ve range value.

Syntax

```
ve range <ID
no ve range
```

Parameters

<ID> – identifier of the edge device, may take values in the range [8...100].

Required privilege level

10

Default value

10

Command mode

CONFIG-AUTODISCOVERY-BGP

Example

```
esr(config-autodiscovery-bgp)# ve range 20
```

vpn id

This command sets the identifier of the VPN connection.

The use of a negative form (no) of the command removes the parameter from the configuration.

Syntax

```
vpn id <ID
no vpn id
```

Parameters

<ID> – VPN connection identifier, takes values in the range of [1..4294967295].

Required privilege level

10

Command mode

CONFIG-AUTODISCOVERY-BGP

Example

```
esr(config-autodiscovery-bgp)# vpn id 5
```

bridge-group

This command is used to specify VSI (Virtual Switching Instance).

The use of a negative form (no) of the command removes the parameter from the configuration.

Syntax

```
bridge-group <INDEX>
no bridge-group
```

Parameters

<INDEX> – bridge-group index. Bridge with the specified index must be added to the configuration beforehand.

Required privilege level

10

Command mode

CONFIG-L2VPN-EOMPLS

Example

```
esr(config-l2vpn-vpls)# bridge-group 25
```

description

This command sets the description.

The use of a negative form (no) of the command removes description.

Syntax

```
description <LINE>
no description
```

Parameters

<LINE> – description. Set by the string [1..255] characters long.

Required privilege level

10

Command mode

CONFIG-L2VPN-PW

CONFIG-L2VPN-PW-CLASS

Example

```
esr(config-l2vpn-pw)# description PW_FOR_VLAN398
```

enable

The command enables a configurable functionality.

The use of a negative form (no) of the command disables the configurable functionality.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters

Required privilege level

10

Command mode

CONFIG-L2VPN-PW

CONFIG-L2VPN-EOMPLS

Example

```
esr(config-l2vpn-pw)# enable
```

encapsulation mpls mtu

This command sets the MTU value for the pseudowire interfaces included in this pw-class.

The use of a negative form (no) of the command sets the default MTU value.

Syntax

```
encapsulation mpls mtu <MTU>
no encapsulation mpls mtu
```

Parameters

<MTU> — MTU value, takes values in the range of [552..10000]

Default value

1500

Required privilege level

10

Command mode

CONFIG-L2VPN-PW-CLASS

Example

```
esr(config-l2vpn-pw-class)# encapsulation mpls mtu 1464
```

encapsulation mpls status-tlv disable

This command disables status-tlv messaging.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] encapsulation mpls status-tlv disable
```

Parameters

The command does not contain parameters.

Default value

status-tlv enable

Required privilege level

10

Command mode

CONFIG-L2VPN-PW-CLASS

Example

```
esr(config-l2vpn-pw-class)# encapsulation mpls status-tlv disable
```

interface

This command is used to set the Attached Circuit interface.

The use of a negative form (no) of the command removes a Attached Circuit interface.

Syntax

```
interface { <IF> | <TUN> }
no interface
```

Parameters

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels;

Required privilege level

10

Command mode

CONFIG-L2VPN-EOMPLS

Example

```
esr(config-l2vpn-p2p)# interface gigabitethernet 1/0/4.398
```

neighbor-address

This command specifies the address of the router to which the pseudo-wire will be established.

The use of a negative form (no) of the command sets the default value.

Syntax

```
neighbor-address <ADDR>
no neighbor-address
```

Parameters

<ADDR> – router IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Default value

The address of the neighbor is taken as LSR_ID

Required privilege level

10

Command mode

CONFIG-L2VPN-EOMPLS

Example

```
esr(config-l2vpn-pw)# neighbor-address 192.168.24.78
```

p2p

This command is used to create a point-to-point type L2VPN (VPWS).

The use of a negative form (no) of the command removes a specified p2p.

Syntax

```
[no] p2p <NAME>
```

Parameters

<NAME> — name of the p2p service, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-L2VPN

Example

```
esr(config-l2vpn)# p2p VLAN_398
```

pw

This command is used to create the pseudo-wire.

The use of a negative form (no) of the command removes the pseudo-wire.

Syntax

```
pw <PW_ID> <LSR_ID>
```

```
no pw
```

Parameters

<PW_ID> — psewdowire identifier, specified in the range [1..4294967295]

<LSR_ID> — identifier of LSR to which pseudo-wire is built, specified as AAA.BBB.CCC.DDD, where each part takes values [0..255]

Required privilege level

10

Command mode

CONFIG-L2VPN-EOMPLS

Example

```
esr(config-l2vpn-p2p)# pw 398 1.1.1.1
```

pw-class

This command is used to create the pw-class.

The use of a negative form (no) of the command removes the pw-class from the configuration.

Syntax

`[no] pw-class <WORD>`

Parameters

<WORD> — pw-class name [1..31] characters long.

Required privilege level

10

Command mode

CONFIG-L2VPN

Example

```
esr(config-l2vpn)# pw-class Class1
```

pw-class

This command is used to set the pw class for the pseudowire.

The use of a negative form (no) of the command removes the pw class for the given pseudowire.

Syntax

`pw-class <WORD>`

`no pw-class`

Parameters

<WORD> — pw-class name [1..31] characters long.

Required privilege level

10

Command mode

CONFIG-L2VPN-PW

Example

```
esr(config-l2vpn-pw)# pw-class Class1
```

transport-mode

This command allows to select the pseudowire transport mode.

The use of a negative form (no) of the command returns the default value

Syntax

```
transport-mode { ethenet | vlan }
no transport-mode
```

Parameters

<ethernet> — mode in which the 802.1Q tag is removed from the header when entering pseudo-wire;

<vlan> — mode in which the 802.1Q tag can be saved when transmitted over pseudo-wire.

Default value

ethernet

Required privilege level

10

Command mode

CONFIG-L2VPN-EOMPLS

Example

```
esr(config-l2vpn-p2p)# transport-mode vlan
```

vpls

This command is used to create a point-to-multipoint type L2VPN.

The use of a negative form (no) of the command removes a specified vpls.

Syntax

```
[no] vpls <NAME>
```

Parameters

<NAME> — name of the vpls service, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-L2VPN

Example

```
(config-l2vpn)# vpls Customer1
```

show mpls l2vpn bindings

This command allows to display information on the pseudowire parameters.

Syntax

```
show mpls l2vpn binding [ { [ local <LABEL-ID> ] [ remote <LABEL ID> ] | neighbor <ADDR>
| pw <PW-ID> | ve-id <VE-ID> } ]
```

Parameters

<LABEL-ID> – MPLS-label, specified in the range [0..104875]

<ADDR> – LDP neighbor's IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<PW-ID> — psewdowire identifier, specified in the range [1..4294967295]

<VE-ID> – identifier of the PE router in the VPLS domain [1.. 16384]

Required privilege level

1

Command mode

ROOT

Example

```
esr# show mpls l2vpn bindings pw 100
Neighbor: 1.1.1.1, PW ID: 100
Local label: 19
Type: Eth Tagged
Group ID: 0
MTU: 1500
Remote label: 28
Type: Eth Tagged
Group ID: 0
MTU: 1500
```

show mpls l2vpn pseudowire

This command allows to display information on the pseudowire status.

Syntax

```
show mpls l2vpn pseudowire [ pw <PW-ID> ] [neighbor <ADDR> ]
```

Parameters

<PW-ID> — psewdowire identifier, specified in the range [1..4294967295]

<ADDR> – targeted LDP neighbor's IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

1

Command mode

ROOT

Example

```
esr# show mpls l2vpn pseudowire
Neighbor                                                         PW ID                    Type
Status
---------------------------------------                        ----------
----------            --------
1.1.1.1                                                            100                 Eth
Tagged                Up
```

show mpls l2vpn p2p

This command allows to get up-to-date information on the VPWS service.

Syntax

```
show mpls l2vpn p2p  [ name <WORD> ]
```

Parameters

<WORD> — service name [1..31] characters long.

Required privilege level

1

Command mode

ROOT

Example

```
esr# sh mpls l2vpn p2p
P2P: VPWS
    gigabitethernet 1/0/4.10:
        MTU:    1500
        Status: Up
    PW ID 100, Neighbor 10.10.0.2:
        MTU:         1500
        Status TLV:  Enable
        Last change: 00:05:35
        Status:      Up
```

show mpls l2vpn pw-class

This command allows to retrieve operational information from the configuration pattern applied to the pseudowire.

Syntax

```
show mpls l2vpn pw-class  [ name <WORD> ]
```

Parameters

<WORD> — pattern name [1..31] characters long.

Required privilege level

1

Command mode

ROOT

Example

```
PE1# sh mpls l2vpn pw-class
PW-class                         Neighbor   PW ID      Status Status-tlv MTU
-------------------------------- ---------- ---------- ------ ---------- ------
PW_template                      10.10.0.2  100        UP    Enable     1500
```

show mpls l2vpn vpls

This command allows to get up-to-date information on the VPLS service.

Syntax

```
show mpls l2vpn vpls [ name <WORD> ]
```

Parameters

<WORD> — service name [1..31] characters long.

Required privilege level

1

Command mode

ROOT

Example

```
esr# sh mpls l2vpn vpls
VPLS: VPLS
    bridge 100:
        MTU:    1500
        Status: UP
    PWs:
      PW ID 23, Neighbor 10.10.0.1:
        MTU:         1500
        Status TLV:  Enable
        Last change: 19:00:07
        Status:      UP
```

25 Redundancy

- VRRP management
    - clear vrrp-state
    - show vrrp
    - vrrp
    - vrrp authentication algorithm
    - vrrp authentication key
    - vrrp force-up
    - vrrp group
    - vrrp id
    - vrrp ip
    - vrrp preempt delay
    - vrrp preempt disable
    - vrrp priority
    - vrrp source-ip
    - vrrp timers advertise
    - vrrp timers garp delay
    - vrrp timers garp refresh
    - vrrp timers garp refresh-repeat
    - vrrp timers garp repeat
    - vrrp track-ip
    - vrrp track-ip interval
    - vrrp track-ip packets
    - vrrp version
- IPv6 VRRP management
    - ipv6 vrrp
    - ipv6 vrrp group
    - ipv6 vrrp id
    - ipv6 vrrp ip
    - ipv6 vrrp preempt delay
    - ipv6 vrrp preempt disable
    - ipv6 vrrp priority
    - ipv6 vrrp source-ip
    - ipv6 vrrp timers advertise
    - ipv6 vrrp timers nd delay
    - ipv6 vrrp timers nd refresh
    - ipv6 vrrp timers nd refresh-repeat
    - ipv6 vrrp timers nd repeat
    - show ipv6 vrrp
- DHCP redundancy configuration
    - ip dhcp-server failover
    - ip dhcp-server failover auto-partner-down
    - ip dhcp-server failover local-address
    - ip dhcp-server failover maximum-client-lead-time
    - ip dhcp-server failover remote-address
    - ip dhcp-server failover role
    - show ip dhcp server failover
- Firewall redundancy configuration
    - ip firewall failover
    - ip firewall failover source-address
    - ip firewall failover destination-address
    - ip firewall failover port
    - ip firewall failover sync-type

- ip firewall failover multicast-address
- ip firewall failover multicast-group
- ip firewall failover vrrp-group
- show ip firewall failover
- show high-availability state

- Dual-Homing management
  - backup interface
  - backup-interface mac-duplicate
  - backup-interface mac-per-second
  - backup-interface preemption
  - show interfaces backup
- MultiWAN configuration
  - description
  - enable
  - extended count
  - extended jitter
  - extended loss
  - extended period
  - extended rtt
  - extended timeout
  - failover
  - ip address
  - ipv6 address
  - ipv6 wan load-balance enable
  - ipv6 wan load-balance failure-count
  - ipv6 wan load-balance nexthop
  - ipv6 wan load-balance rule
  - ipv6 wan load-balance source-address
  - ipv6 wan load-balance success-count
  - ipv6 wan load-balance target-list
  - ipv6 wan load-balance target-list
  - ipv6 wan load-balance target-list check-all
  - mode
  - outbound
  - resp-time
  - target
  - wan load-balance enable
  - wan load-balance failure-count
  - wan load-balance nexthop
  - wan load-balance rule
  - wan load-balance source-address
  - wan load-balance success-count
  - wan load-balance target-list
  - wan load-balance target-list
  - wan load-balance target-list check-all
  - show wan rules
  - show wan cellular status modem
  - show wan interfaces status
  - show wan tunnels status
  - show ipv6 wan rules
  - show ipv6 wan interfaces status

VRRP management

clear vrrp-state

This command stops VRRP protocol execution for time 3 * Advertisement_Interval +1. This enables the router in the backup state to perform a master hijacking.

Syntax

```
clear vrrp-state
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example 1

```
esr# clear vrrp-state
```

show vrrp

This command displays information about VRRP.

Syntax

```
show vrrp [ <ID> ] [ vrf <VRF> ]
```

Parameters

<ID> – process number, takes values of [1..255];

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example 1

```
esr# show vrrp
Virtual router   Virtual IP          Priority   Preemption   State
--------------   ----------------    --------   ----------   ------
4                4.4.4.1             100        Enabled      Master
```

Example 2

```
esr# show vrrp 4
Interface                  bridge 50
State:                     Master
Virtual IP address:        4.4.4.1
Source IP address:         4.4.4.4
Virtual MAC address:       00:00:5e:00:01:04
Advertisement interval:    1
Preemption:                Enabled
Priority:                  100
Synchronization group ID: --
```

vrrp

This command enables VRRP process on IP interface.

The use of a negative form (no) of the command disables VRRP process.

Syntax

```
[no] vrrp
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# vrrp
```

vrrp authentication algorithm

This command defines authentication algorithm.

The use of a negative form (no) of the command disables the authentication.

Syntax

```
vrrp authentication algorithm <ALGORITHM>
no vrrp authentication algorithm
```

Parameters

<ALGORITHM> – authentication algorithm:

- cleartext – password, transmitted in clear text;
- md5 – password is hashed by md5 algorithm.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-gre)# vrrp authentication algorithm cleartext
```

vrrp authentication key

This command sets a password for authentication with a neighbor.

The use of a negative form (no) of the command removes the password.

Syntax

```
vrrp authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no vrrp authentication key
```

Parameters

<CLEAR-TEXT> – password of 8 to 16 bytes;
<ENCRYPTED-TEXT> – encrypted password of 8 to 16 bytes (from 16 to 32 characters) in hexadecimal format (0xYYYY ...) or (YYYY ...).

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# vrrp authentication key ascii-text 123456789
esr(config-if-gi)# vrrp authentication key ascii-text encrypted CDE65039E5591FA3F1
```

vrrp force-up

This command sets the mode, in which the virtual IP interface remains in the UP state regardless of the state of the interface itself.

The use of a negative form (no) of the command sets the default mode.

Syntax

```
[no] vrrp force-up
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# vrrp force-up
```

vrrp group

This command sets the VRRP interface to the group. The group provides the ability to synchronize several VRRP processes, so if in one of the processes there is a wizard change, then in another process roles will also be changed.

The use of a negative form (no) of the command removes VRRP router from the group.

Syntax

```
vrrp group <GRID>
```

```
no vrrp group
```

Parameters

<GRID> – VRRP router group identifier, takes values in the range of [1..32].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# vrrp group 10
```

vrrp id

This command sets VRRP router identifier.

The use of a negative form (no) of the command removes virtual router identifier.

Syntax

```
vrrp id <VRID>
no vrrp id
```

Parameters

<VRID> – VRRP router identifier, takes values in the range of [1..255].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# vrrp id 125
```

vrrp ip

This command sets the virtual IP address of the VRRP router.

The use of a negative form (no) of the command removes the virtual IP address of the router.

Syntax

```
vrrp ip <ADDR/LEN > [ secondary ]
no vrrp ip
```

Parameters

<ADDR/LEN> – virtual IP address and mask length, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32]. You can specify several IP addresses separated by commas. Up to 8 IP addresses can be assigned to the interface.

secondary – key to set an additional IP address.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-bridge)# vrrp ip 115.0.0.1
```

vrrp preempt delay

This command sets the time interval after which a Backup router with a higher priority will attempt to pickup the Master role for the current Master router with a lower priority.

The use of a negative form (no) of the command returns the default configuration.

Syntax

```
[no] vrrp preempt delay <TIME>
no vrrp preempt delay
```

Parameters

<TIME> – timeout, takes value in seconds [1..1000].

Default value

0

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# vrrp preempt delay 100
```

vrrp preempt disable

This command defines whether a Backup router with a higher priority will attempt to pickup the Master role for the current Master router with a lower priority.

An exception is the router, in which the virtual IP address matches the IP address on the interface, it will always take over the Master role regardless of this setting.

The use of a negative form (no) of the command returns the default configuration.

Syntax

`[no] vrrp preempt disable`

Parameters

The command does not contain parameters.

Default value

Switching enabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# vrrp preempt disable
```

vrrp priority

This command sets VRRP router priority.

The use of a negative form (no) of the command sets the default priority value.

Syntax

```
vrrp priority <PR>
no vrrp priority
```

Parameters

<PR> – VRRP router priority, takes values in the range of [1..254].

Default value

100

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# vrrp priority 189
```

vrrp source-ip

This command sets the IP address that will be used as the sender's IP address for VRRP messages.

The use of a negative form (no) of the command removes a specified IP address.

Syntax

```
vrrp source-ip <IP>
no vrrp source-ip
```

Parameters

<IP> – sender IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-bridge)# vrrp source-ip 115.0.0.10
```

vrrp timers advertise

This command defines the time interval between sending VRRP messages.

The use of a negative form (no) of the command restores the default value.

Syntax

```
vrrp timers advertise <TIME>
no vrrp timers advertise
```

Parameters

<TIME> – time in seconds, takes values of [1..40].

Default value

1

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-gre)# vrrp timers advertise 4
```

vrrp timers garp delay

This command defines the interval after which the Gratuituous ARP message(s) are sent when the router enters the Master state.

The use of a negative form (no) of the command restores the default value.

Syntax

```
vrrp timers garp delay <TIME>
no vrrp timers garp delay
```

Parameters

<TIME> – time in seconds, takes values of [1..60].

Default value

5

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-gre)# vrrp timers garp delay 4
```

vrrp timers garp refresh

This command defines the interval after which the Gratuituous ARP message(s) will be sent while the router is in the Master state.

The use of a negative form (no) of the command restores the default value.

Syntax

```
vrrp timers garp refresh <TIME>
no vrrp timers garp refresh
```

Parameters

<TIME> – time in seconds, takes values of [1..65535].

Default value

Periodic sending is disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-gre)# vrrp timers garp refresh 4
```

vrrp timers garp refresh-repeat

This command defines the number of Gratuituous ARP messages that will be sent with a **garp refresh** period while the router is in the Master state.

The use of a negative form (no) of the command restores the default value.

Syntax

```
vrrp timers garp refresh-repeat <COUNT>
```

```
no vrrp timers garp refresh-repeat
```

Parameters

<COUNT> – amount of messages, takes values of [1..60].

Default value

1

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-gre)# vrrp timers garp refresh-repeat 10
```

vrrp timers garp repeat

This command defines the number of Gratuituous ARP messages that will be sent when the router transitions to the Master state.

The use of a negative form (no) of the command restores the default value.

Syntax

```
vrrp timers garp repeat <COUNT>
```

```
no vrrp timers garp repeat
```

Parameters

<COUNT> – amount of messages, takes values of [1..60].

Default value

5

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-gre)# vrrp timers garp repeat 10
```

vrrp track-ip

This command enables vrrp status management based on icmp-reply from a specific IP address.

The use of a negative form (no) of the command disables vrrp status management based on icmp-reply from a specific IP address.

Syntax

```
[no] vrrp track-ip <ADDR>
```

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if)# vrrp track-ip 192.168.154.22
```

vrrp track-ip interval

This command sets the check interval using icmp-request to manage the vrrp status based on icmp-reply from a specific IP address.

The use of a negative form (no) of the command sets the default interval value.

Syntax

```
vrrp track-ip interval <TIME>
no vrrp track-ip interval
```

Parameters

<TIME> – icmp-request send interval in seconds [3..60].

Default value

10

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-gre)# vrrp track-ip interval 20
```

vrrp track-ip packets

This command sets the number of icmp-request at each check to manage the vrrp status based on icmp-reply from a specific IP address.

The use of a negative form (no) of the command sets the default value of number of packets.

Syntax

```
vrrp track-ip packets <COUNT>
```

```
no vrrp track-ip packets
```

Parameters

<COUNT> – icmp-request send interval in seconds [1..5]

Default value

5

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-gre)# vrrp track-ip packets 3
```

vrrp version

The command sets VRRP version.

The use of a negative form (no) of the command sets the default value.

Syntax

```
vrrp version <VERSION>
```

```
no vrrp version
```

Parameters

<VERSION> – VRRP version: 2, 3.

Default value

2

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-gre)# vrrp version 3
```

IPv6 VRRP management

ipv6 vrrp

This command enables VRRP process on IPv6 interface.

The use of a negative form (no) of the command disables VRRP process.

Syntax

```
[no] ipv6 vrrp
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 vrrp
```

ipv6 vrrp group

This command sets the VRRP router to the group. The group provides the ability to synchronize several VRRP processes, so if in one of the processes there is a wizard change, then in another process roles will also be changed.

The use of a negative form (no) of the command removes VRRP router from the group.

Syntax

`ipv6 vrrp group <GRID>`

`no ipv6 vrrp group`

Parameters

<GRID> – VRRP router group identifier, takes values in the range of [1..32].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 vrrp group 10
```

ipv6 vrrp id

This command sets VRRP router identifier.

The use of a negative form (no) of the command removes virtual router identifier.

Syntax

```
ipv6 vrrp id <VRID>
no ipv6 vrrp id
```

Parameters

<VRID> – VRRP router identifier, takes values in the range of [1..255].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 vrrp id 125
```

ipv6 vrrp ip

This command sets the virtual IPv6 address of the VRRP router.

The use of a negative form (no) of the command removes the virtual IPv6 address of the router.

Syntax

```
ipv6 vrrp ip <IPV6-ADDR> [secondary]
no ipv6 vrrp ip
```

Parameters

<IPV6-ADDR> – virtual IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF]. You can specify several IPv6 addresses separated by commas. Up to 8 IPv6 addresses can be assigned to the interface.

secondary – key to set an additional IP address.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-bridge)# ipv6 vrrp ip fc00::1
```

ipv6 vrrp preempt delay

This command sets the time interval after which a Backup router with a higher priority will attempt to pickup the Master role for the current Master router with a lower priority.

The use of a negative form (no) of the command returns the default configuration.

Syntax

```
ipv6 vrrp preempt delay <TIME>
no ipv6 vrrp preempt delay
```

Parameters

<TIME> – timeout, takes value in seconds [1..1000].

Default value

0

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 vrrp preempt delay 100
```

ipv6 vrrp preempt disable

This command defines whether a Backup router with a higher priority will attempt to pickup the Master role for the current Master router with a lower priority.

An exception is the router, in which the virtual IPv6 address matches the IPv6 address on the interface, it will always take over the Master role regardless of this setting.

The use of a negative form (no) of the command returns the default configuration.

Syntax

```
[no] ipv6 vrrp preempt disable
```

Parameters

The command does not contain parameters.

Default value

Switching enabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 vrrp preempt disable
```

ipv6 vrrp priority

This command sets VRRP router priority.

The use of a negative form (no) of the command sets the default priority value.

Syntax

```
ipv6 vrrp priority <PR>
no ipv6 vrrp priority
```

Parameters

<PR> – VRRP router priority, takes values in the range of [1..254].

Default value

100

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 vrrp priority 189
```

ipv6 vrrp source-ip

This command sets the IPv6 address that will be used as the sender's IPv6 address for VRRP messages.

The use of a negative form (no) of the command removes a specified IPv6 address.

Syntax

```
ipv6 vrrp source-ip <IPV6-ADDR>
no ipv6 vrrp source-ip
```

Parameters

<IPV6-ADDR> – sender IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-bridge)# ipv6 vrrp source-ip fc00::2
```

ipv6 vrrp timers advertise

This command defines the time interval between sending VRRP messages.

The use of a negative form (no) of the command restores the default value.

Syntax

```
ipv6 vrrp timers advertise <TIME>
no ipv6 vrrp timers advertise
```

Parameters

<TIME> – time in seconds, takes values of [1..40].

Default value

1

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 vrrp timers advertise 4
```

ipv6 vrrp timers nd delay

This command defines the delay between the establishment of the IPv6 VRRP MASTER state and the start of the distribution of ND messages.

The use of a negative form (no) of the command restores the default value.

Syntax

```
ipv6 vrrp timers nd delay <TIME>
no ipv6 vrrp timers nd delay
```

Parameters

<TIME> – time in seconds, takes values of [1..60].

Default value

5

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 vrrp timers nd delay 30
```

ipv6 vrrp timers nd refresh

This command defines the update period of the ND protocol information for IPv6 VRRP in the MASTER state.

The use of a negative form (no) of the command restores the default value.

Syntax

```
ipv6 vrrp timers nd refresh <TIME>
no ipv6 vrrp timers nd refresh
```

Parameters

<TIME> – time in seconds, takes values of [1..65535].

Default value

5

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 vrrp timers nd refresh 32768
```

ipv6 vrrp timers nd refresh-repeat

This command defines the number of ND messages sent over the update period for an IPv6 VRRP in the MASTER state.

The use of a negative form (no) of the command restores the default value.

Syntax

```
ipv6 vrrp timers nd refresh-repeat <NUM>
no ipv6 vrrp timers nd refresh-repeat
```

Parameters

<NUM> – amount, takes values of [1..60].

Default value

0

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 vrrp timers nd refresh-repeat 12
```

ipv6 vrrp timers nd repeat

This command defines the number of ND packets sent after setting IPv6 VRRP to the MASTER state.

The use of a negative form (no) of the command restores the default value.

Syntax

```
ipv6 vrrp timers nd repeat <NUM>
no ipv6 vrrp timers nd repeat
```

Parameters

<NUM> – amount, takes values of [1..60].

Default value

1

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 vrrp timers nd repeat 4
```

show ipv6 vrrp

This command displays information about VRRP.

Syntax

```
show ipv6 vrrp [ <ID> ] [ vrf <VRF> ]
```

Parameters

<ID> – process number, takes values of [1..255];

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example 1

```
esr# show ipv6 vrrp
Virtual router   Virtual IP                                              Priority   Preemption
State
--------------   -----------------------------------------------------   --------   ----------
------
2                fc00::3                                                 100        Disabled
Master
```

Example 2

```
esr# show ipv6 vrrp 2
Interface                 gi1/0/1
State:                    Master
Source IP address:        fe80::aaf9:4bff:feaa:3a1
Virtual IP address:       fc00::3
Virtual MAC address:      00:00:5e:00:01:02
Advertisement interval:   1
Preemption:               Disabled
Priority:                 100
Synchronization group ID: --
```

DHCP redundancy configuration

ip dhcp-server failover

This command enables DHCP server redundancy. The use of a negative form (no) of the command disables DHCP server redundancy.

Syntax

```
[no] ip dhcp-server failover
```

Parameters

The command does not contain parameters.

Default value

Disabled

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp-server failover
```

ip dhcp-server failover auto-partner-down

This command sets the DHCP-lease run time at which a lease can be extended by one of the fault tolerance hosts without contacting the other.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip dhcp-server failover auto-partner-down <TIME>
no ip dhcp-server failover auto-partner-down
```

Parameters

<TIME> – time, takes value in seconds [1..86400].

Default value

0

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp-server failover auto-partner-down 200
```

ip dhcp-server failover local-address

This command sets the IP address, port (TCP 647), on which the DHCP server receives Failover messages when operating in redundant mode.

The use of a negative form (no) of the command removes set IP address.

Syntax

```
ip dhcp-server failover local-address <ADDR>
no ip dhcp-server failover local-address
```

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp-server failover local-address 192.168.1.1
```

ip dhcp-server failover maximum-client-lead-time

This command sets the maximum time for which the DHCP server can extend the lease time of the IP address.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip dhcp-server failover maximum-client-lead-time <TIME>
no ip dhcp-server failover maximum-client-lead-time
```

Parameters

<TIME> – time, takes value in seconds [1..86400].

Default value

1800

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp-server failover maximum-client-lead-time 200
```

ip dhcp-server failover remote-address

This command sets the IP address of the backup DHCP server to which the failover messages are sent when working in redundancy mode.

The use of a negative form (no) of the command removes redundant DHCP server IP address.

Syntax

```
ip dhcp-server failover remote-address <ADDR>
no ip dhcp-server failover remote-address
```

Parameters

<ADDR> – redundant DHCP server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp-server failover remote-address 192.168.1.2
```

ip dhcp-server failover role

This command defines the role of the DHCP server when operating in redundant mode.

The use of a negative form (no) of the command removes the role of the DHCP server when operating in redundant mode.

Syntax

```
ip dhcp-server failover role <ROLE>
no ip dhcp-server failover role
```

Parameters

<ROLE> – role of the DHCP server when operating in redundant mode:

- primary – active DHCP server mode;
- secondary – redundant DHCP server mode.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp-server failover role primary
```

show ip dhcp server failover

This command displays status of DHCP server redundancy.

Syntax

```
show ip dhcp server failover
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip dhcp server failover

Status:                                      Disabled
```

Firewall redundancy configuration

ip firewall failover

This command enables Firewall sessions redundancy.

The use of a negative form (no) of the command disables Firewall sessions redundancy.

Syntax

```
[no] ip firewall failover
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip firewall failover
```

ip firewall failover source-address

This command sets the IP address of the network interface from which messages will be sent when the Firewall is running in session redundancy mode.

The use of a negative form (no) of the command removes IP address of the outgoing interface.

Syntax

```
ip firewall failover source-address <ADDR>
no ip firewall failover source-address
```

Parameters

<ADDR> – network interface IP address from which messages will be sent, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall failover source-address 192.168.1.1
```

ip firewall failover destination-address

This command sets the IP address of the neighbor when the Firewall session redundancy are running in unicast mode.

The use of a negative form (no) of the command removes neighbor IP address.

Syntax

```
ip firewall failover destination-address <ADDR>
no ip firewall failover destination-address
```

Parameters

<ADDR> – neighbor’s IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip firewall failover destination-address 192.168.1.2
```

ip firewall failover port

This command sets the UDP port number of the Firewall session redundancy service, through which information is exchanged when working in unicast mode.

The use of a negative form (no) of the command removes Firewall session redundancy service port number.

Syntax

```
ip firewall failover port <PORT>
no ip firewall failover port
```

Parameters

<PORT> – port number of the Firewall session redundancy service, may take values [1..65535].

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall failover port 3333
```

ip firewall failover sync-type

This command defines the mode of information exchange between the primary and redundant routers.

The use of a negative form (no) of the command removes Firewall redundancy operation mode.

Syntax

```
ip firewall failover sync-type <MODE>
```

```
no ip firewall failover sync-type
```

Parameters

<MODE> – information exchange mode:

- unicast – unicast mode;
- multicast – multicast mode.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip firewall failover sync-type multicast
```

ip firewall failover multicast-address

This command sets the multicast IP address that will be used to exchange information when working on the redundancy of Firewall sessions in multicast mode.

The use of a negative form (no) of the command removes multicast IP address.

Syntax

```
ip firewall failover multicast-address <ADDR>
```

```
no ip firewall failover multicast-address
```

Parameters

<ADDR> – multicast IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip firewall failover multicast-address 238.0.0.10
```

ip firewall failover multicast-group

This command sets the multicast group identifier for information exchange when working on the redundancy of Firewall sessions in multicast mode.

The use of a negative form (no) of the command removes the group identifier.

Syntax

```
ip firewall failover multicast-group <GROUP>
no ip firewall failover multicast-group
```

Parameters

<GROUP> – multicast group, may take values in range [1000..9999].

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip firewall multicast-group 1028
```

ip firewall failover vrrp-group

This command defines the VRRP group, on the basis of which the state (master/redundant) of the router is defined when reserving Firewall sessions.

The use of a negative form (no) of the command removes the VRRP group identifier.

Syntax

```
ip firewall failover vrrp-group <GRID>
no ip firewall failover vrrp-group
```

Parameters

<GRID> – VRRP router group identifier, takes values in the range of [1..32].

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip firewall failover vrrp-group 10
```

show ip firewall failover

This command displays Firewall sessions redundancy status.

Syntax

```
show ip firewall failover
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip firewall failover
Communication interface:                     br6
Status:                                      Running
Bytes sent:                                  6407688
Bytes received:                              6355040
Packets sent:                                430149
Packets received:                            429844
Send errors:                                 0
Receive errors:                              0
```

show high-availability state

This command allows you to view the overall state of the redundancy systems and the role of the device.

Syntax

`show high-availability state`

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show high-availability state
VRRP role:                         Master
AP Tunnels:
    State:                         successful synchronization
    Last synchronization:          17:22:11 08.06.2015
DHCP server:
    State:                         successful synchronization
    Last state change:             17:49:42 03.06.2015
Firewall sessions:
    State:                         successful synchronization
    Last synchronization:          17:22:18 08.06.2015
```

Dual-Homing management

⚠ In the current firmware version, this functionality is supported only by ESR-1000 router.

backup interface

This command specifies the redundancy interface to which the switch will occur when communication is lost on the primary. Redundancy is enabled only on those interfaces on which the Spanning Tree protocol is disabled and VLAN Ingress Filtering is enabled.

The use of a negative form (no) of the command removes configuration from the interface.

Syntax

`backup interface <IF> vlan <VID>`

`no backup interface`

Parameters

<IF> – an interface, specified in the form described in Section Types and naming order of router interfaces.

<VID> – VLAN ID, set in the range of [2..4094]. You can also specify it by the range with '-' or by comma-separated list.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# backup interface gigabitethernet 1/0/15 vlan 10-200
```

backup-interface mac-duplicate

This command defines the number of copies of packets with the same MAC address that will be sent to the active interface when switching.

The use of a negative form (no) of the command restores the default value.

Syntax

```
backup-interface mac-duplicate <COUNT>
no backup-interface mac-duplicate
```

Parameters

<COUNT> – packet copies amount, take values [1..4].

Default value

1

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# backup-interface mac-duplicate 4
```

backup-interface mac-per-second

This command defines the number of packets per second that will be sent to the active interface when switching.

The use of a negative form (no) of the command restores the default value.

Syntax

```
backup-interface mac-per-second <COUNT>
```

```
no backup-interface mac-per-second
```

Parameters

<COUNT> – number of packets per second, takes values [50..400].

Default value

400

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# backup-interface mac-per-second 200
```

backup-interface preemption

This command indicates that it is necessary to switch to the main interface when restoring communication. If the recovery of the main interface is configured while the backup is active, then when the link is raised on the main interface, the traffic will be switched to it.

The use of a negative form (no) of the command restores the default configuration.

Syntax

```
[no] backup-interface preemption
```

Parameters

The command does not contain parameters.

Default value

Switch disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# backup-interface preemption
```

show interfaces backup

This command displays information about the status of the main and backup interfaces.

Syntax

`show interfaces backup`

Required privilege level

1

Command mode

ROOT

Example

```
esr# show interfaces backup
   Backup Interface Options:
      Preemption is disabled.
      MAC recovery packets rate 400 pps.
      Recovery packets repeats count 1.
VID    Master Interface            Backup Interface            State
----   -------------------------   -------------------------   ------------------------------
10     gigabitethernet 1/0/3       gigabitethernet 1/0/4       Master Up/Backup Down
----   -------------------------   -------------------------   ------------------------------
11     gigabitethernet 1/0/3       gigabitethernet 1/0/4       Master Up/Backup Down
----   -------------------------   -------------------------   ------------------------------
12     gigabitethernet 1/0/3       gigabitethernet 1/0/4       Master Up/Backup Down
```

MultiWAN configuration

description

This command defines rule description.

The use of a negative form (no) of the command removes description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – wan rule description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-WAN-RULE

CONFIG-IPV6-WAN-RULE

CONFIG-WAN-TARGET

CONFIG-IPV6-WAN-TARGET

Example

```
esr(config-wan-rule)# description "tunnel to branch"
```

enable

This command enables wan rule, check of the target.

The use of negative form (no) of the command disables wan rule, check of the target.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-WAN-RULE

CONFIG-WAN-TARGET

CONFIG-IPV6-WAN-RULE

CONFIG-IPV6-WAN-TARGET

Example

Target check:

```
esr(config-ipv6-wan-rule)# enable
```

extended count

This command defines number of simultaneously sent ICMP requests.

The use of a negative form (no) of the command sets the default value.

Syntax

`extended count <NUM>`

`no extended count`

Parameters

<NUM> – number of simultaneously sent ICMP requests, takes values in range [10..100].

Default value

10

Required privilege level

10

Command mode

CONFIG-WAN-TARGET

Example

```
esr(config-wan-target)# extended count 75
```

extended jitter

This command defines the jitter threshold value, above which the channel quality is unsatisfactory.

The use of a negative form (no) of the command sets the default value.

Syntax

`extended jitter <TIME>`

```
no extended jitter
```

Parameters

<TIME> – the jitter threshold in milliseconds, takes value in the range [50-1000].

Default value

50

Required privilege level

10

Command mode

CONFIG-WAN-TARGET

Example

```
esr(config-wan-target)# extended jitter 78500
```

extended loss

This command defines the loss percent threshold value, above which the channel quality is unsatisfactory.

The use of a negative form (no) of the command sets the default value.

Syntax

```
extended loss <NUM>
no extended loss
```

Parameters

<NUM> – loss percent threshold [1..100]

Default value

1

Required privilege level

10

Command mode

CONFIG-WAN-TARGET

Example

```
esr(config-wan-target)# extended loss 80
```

extended period

This command defines the period of sending ICMP requests.

The use of a negative form (no) of the command sets the default value.

Syntax

```
extended period <TIME>

no extended period
```

Parameters

<TIME> – period of ICMP requests sending in seconds, takes a value in the range [10..600].

Default value

240

Required privilege level

10

Command mode

CONFIG-WAN-TARGET

Example

```
esr(config-wan-target)# extended period 120
```

extended rtt

This command defines maximal radial delay threshold value, above which the channel quality is unsatisfactory.

The use of a negative form (no) of the command sets the default value.

Syntax

```
extended rtt <TIME>

no extended rtt
```

Parameters

<TIME> – maximal RTT threshold in milliseconds, takes value in the range [200..1000].

Default value

240

Required privilege level

10

Command mode

CONFIG-WAN-TARGET

Example

```
esr(config-wan-target)# extended rtt 800
```

extended timeout

This command defines maximal waiting time for a response to an ICMP request.

The use of a negative form (no) of the command sets the default value.

Syntax

```
extended timeout <TIME>
no extended timeout
```

Parameters

<TIME> – maximum waiting time for a response to an ICMP request in milliseconds, takes a vlue in the range [500..10000].

Default value

1000

Required privilege level

10

Command mode

CONFIG-WAN-TARGET

Example

```
esr(config-wan-target)# extended timeout 5000
```

failover

This command switchs from balance mode to redundancy mode.

The use of a negative form (no) of the command returns balance mode.

Syntax

```
[no] failover
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-WAN-RULE

CONFIG-IPV6-WAN-RULE

Example

```
esr(config-wan-rule)# failover
```

ip address

This command sets IP address of the check.

The use of a negative form (no) of the command removes a specified address.

Syntax

```
ip address <ADDR>
no ip address
```

Parameters

<ADDR> – destination IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-WAN-TARGET

Example

```
esr(config-wan-target)# ip address 18.168.1.25
```

ipv6 address

This command sets IPv6 address of the check.

The use of a negative form (no) of the command removes a specified address.

Syntax

```
ipv6 address <IPV6-ADDR>
no ipv6 address
```

Parameters

<IPV6-ADDR> – destination IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

10

Command mode

CONFIG-IPV6-WAN-TARGET

Example

```
esr(config-ipv6-wan-target)# ipv6 address fc00::2
```

ipv6 wan load-balance enable

This command enables WAN mode on the interface for IPv6 stack.

The use of a negative form (no) of the command disables WAN mode on the interface for IPv6 stack.

Syntax

```
[no] ipv6 wan load-balance enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 wan load-balance enable
```

ipv6 wan load-balance failure-count

This command defines the number of unsuccessful attempts to check the connection through the IPv6 stack, after which, in the absence of a response from the opposite side, the connection is considered inactive.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 wan load-balance failure-count <VALUE>
no ipv6 wan load-balance failure-count
```

Parameters

<VALUE> – number of attempts, takes values in the range of [1..10].

Default value

5

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 wan load-balance failure-count 3
```

ipv6 wan load-balance nexthop

Set a neighbor's IPv6 address that will be indicated as one of the gateways in a static route created by MultiWAN service.

The use of a negative form (no) of the command removes a specified neighbor IPv6 address.

Syntax

```
ipv6 wan load-balance nexthop <IPV6-ADDR>
no ipv6 wan load-balance nexthop
```

Parameters

<IPV6-ADDR> – destination (gateway) IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

Example

```
esr(config-if-gi)#ipv6 wan load-balance nexthop 220::71
```

ipv6 wan load-balance rule

This command creates WAN rule and switch to the rule for IPv6 parameters configuration mode.

The use of a negative form (no) of the command removes a specified WAN rule.

Syntax

```
ipv6 wan load-balance rule <ID>
no ipv6 wan load-balance rule { <ID> | all }
```

Parameters

<ID> – identifier of the rule being created, may take values:

- For ESR-10/12V/12VF/14VF – [1..25];
- For ESR-20/21/100/200/1000/1200/1500/1511/1700/3100 – [1..50].

The value 'all' is used when deleting all WAN rules for IPv6.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 wan load-balance rule 1
```

ipv6 wan load-balance source-address

This command defines the IPv6 address of the router that will be used as the source IPv6 address in ICMPv6 sent packets for verification purposes.

The use of a negative form (no) of the command removes a specified source IPv6 address.

Syntax

```
ipv6 wan load-balance source-address <IPV6-ADDR>
no ipv6 wan load-balance source-address
```

Parameters

<IPV6-ADDR> – source IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-LT

Example

```
esr(config-if-gi)#ipv6 wan load-balance source-address 220::71
```

ipv6 wan load-balance success-count

Set the amount of successful attempts to check the IPv6 connection, after which, if successful, the connection is considered to be active again.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 wan load-balance success-count <VALUE>
```

```
no ipv6 wan load-balance success-count
```

Parameters

<VALUE> – number of attempts, takes values in the range of [1..10].

Default value

1

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

Example

```
esr(config-if-gi)# ipv6 wan load-balance success-count 3
```

ipv6 wan load-balance target-list

These command creates a list of IPv6 addresses to check the connection integrity and performs the switching to the list parameters configuration mode.

The use of a negative form (no) of the command removes a created list.

Syntax

`ipv6 wan load-balance target-list <NAME>`

`no ipv6 wan load-balance target-list { <NAME> | all }`

Parameters

<NAME> – list name, set by the string of up to 31 characters. The value 'all' is used when deleting all lists of IPv6 addresses to verify the integrity of the connection.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 wan load-balance target-list ten1
```

ipv6 wan load-balance target-list

This command binds the list of IPv6 addresses to verify the integrity of the connection on the network interface.

The use of a negative form (no) of the command removes the list from the interface.

Syntax

```
ipv6 wan load-balance target-list <NAME>
```

```
no ipv6 wan load-balance target-list (удаление привязки списка в режиме конфигурирования сетевых интерфейсов)
```

Parameters

<NAME> – list name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

Example

```
esr(config)# ipv6 wan load-balance target-list ten1
```

ipv6 wan load-balance target-list check-all

This command checks the IPv6 addresses from the integrity check list.

The use of a negative form (no) of the command cancells the check of the IPv6 addresses from the integrity check list. If one of the nodes being checked is unavailable, the gateway will be considered to be unavailable.

Syntax

```
[no] ipv6 wan load-balance target-list check-all
```

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

Example

```
esr(config-if-te)# ipv6 wan load-balance target-list check-all
```

mode

This command defines the target availability check mode, to check the channel status.

The use of a negative form (no) of the command sets the default value.

Syntax

`mode { basic | extended }`

`no mode`

Parameters

basic – basic method, regulated by a valid value of positive responses to ICMP requests;

extended – extended method, it estimates the level of losses, delays, jitter.

Default value

basic

Required privilege level

10

Command mode

CONFIG-WAN-TARGET

Example

```
esr(config-wan-target)# mode extended
```

outbound

This command defines the interfaces or tunnels that are gateways in the route created by the MultiWAN service. The number of gateways in the route depends on the MultiWAN operation mode:

when balancing, the IP addresses of the gateways (section wan load-balance nexthop) of all active interfaces are added to the nexthop route list;

when reserving, the IP address of the gateway (section wan load-balance nexthop) of the active interface with the highest weight is selected as the nexthop route.

The use of a negative form (no) of the command excludes specified interface or tunnel from the MultiWAN rule.

Syntax

```
[no] outbound { interface <IF> | tunnel <TUN> } [<WEIGHT> ]
```

Parameters

<IF> – an interface, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels (only for IPv4);

<WEIGHT> – weight of the tunnel or interface, defines in range [1..255]. If you set the value to 2, then this interface will transmit 2 times more traffic than the interface with the default value. In redundancy mode, the route with the highest weight will be active.

Default value

WEIGHT – 1

Required privilege level

10

Command mode

CONFIG-WAN-RULE

CONFIG-IPV6-WAN-RULE

Example 1

```
esr(config-wan-rule)# outbound interface gigabitethernet 1/0/15
```

Example 2

```
esr(config-ipv6-wan-rule)# outbound interface bridge 2
```

resp-time

This command defines response waiting time to a request via the ICMP.

The use of a negative form (no) of the command sets the default value.

Syntax

```
resp-time <TIME>
no resp-time
```

Parameters

<TIME> – timeout, takes value in seconds [1..30].

Default value

5

Required privilege level

10

Command mode

CONFIG-WAN-TARGET

CONFIG-IPV6-WAN-TARGET

Example

```
esr(config-wan-target)# resp-time 3
```

target

This command creates check target and switch to the target parameters configuration mode.

The use of a negative form (no) of the command removes created target.

Syntax

```
target <ID>
no target { <ID> | all }
```

Parameters

<ID> – target identifier, defines in range [1..50]. If the 'all' parameter value is used when removing, all targets for the configured target list will be removed.

all – delete all targets.

Required privilege level

10

Command mode

CONFIG-WAN-TARGET-LIST

CONFIG-IPV6-WAN-TARGET-LIST

Example

```
esr(config-target-list)# target 1
```

wan load-balance enable

This command enables WAN mode on the interface for IPv4 stack.

The use of a negative form (no) of the command disables WAN mode on the interface for IPv4 stack.

Syntax

```
[no] wan load-balance enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

Example

```
esr(config-if-gi)# wan load-balance enable
```

wan load-balance failure-count

This command defines the number of unsuccessful attempts to check the connection, after which, in the absence of a response from the opposite side, the connection is considered inactive.

The use of a negative form (no) of the command sets the default value.

Syntax

```
wan load-balance failure-count <VALUE>
no wan load-balance failure-count
```

Parameters

<VALUE> – number of attempts, takes values in the range of [1..10].

Default value

5

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

Example

```
esr(config-if-gi)# wan load-balance failure-count 3
```

wan load-balance nexthop

This command sets a neighbor's IP address that will be indicated as one of the gateways in a static route created by MultiWAN service.

The use of a negative form (no) of the command removes a specified neighbor IP address.

Syntax

```
wan load-balance nexthop { <ADDR> | dhcp enable | tunnel enable }
```

```
no wan load-balance nexthop
```

Parameters

<IP> – destination IP address (gateway), defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

dhcp enable – if on the interface an IP address is obtained through a DHCP client, the gateway from the DHCP server is used.

tunnel enable – use peer address as nexthop. Applicable for the interfaced being connected that operate via ppp.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

Example

```
esr(config-gre)# wan load-balance nexthop 16.168.1.25
```

wan load-balance rule

This command creates WAN rule and switch to the rule parameters configuration mode.

The use of a negative form (no) of the command removes a specified WAN rule.

Syntax

```
wan load-balance rule <ID>
no wan load-balance rule { <ID> | all }
```

Parameters

<ID> – identifier of the rule being created, may take values:

- For ESR-10/12V/12VF/14VF – [1..25];
- For ESR-20/21/100/200/1000/1200/1500/1511/1700/3100 – [1..50].

The value 'all' is used when deleting all WAN rules for IPv6.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# wan load-balance rule 1
```

wan load-balance source-address

This command defines the IP address of the router that will be used as the source IP address in ICMP sent packets for verification purposes.

The use of a negative form (no) of the command removes a specified source IP address.

Syntax

```
wan load-balance source-address <ADDR>
no wan load-balance source-address
```

Parameters

<ADDR> – source IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-LT

Example

```
esr(config-gre)# wan load-balance source-address 16.168.1.25
```

wan load-balance success-count

Set the amount of successful attempts to check the connection, after which, if successful, the connection is considered to be active again.

The use of a negative form (no) of the command sets the default value.

Syntax

```
wan load-balance success-count <VALUE>
no wan load-balance success-count
```

Parameters

<VALUE> – number of attempts, takes values in the range of [1..10].

Default value

1

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

Example

```
esr(config-if-gi)# wan load-balance success-count 3
```

wan load-balance target-list

These command creates a list of IP addresses to check the connection integrity and performs the switching to the list parameters configuration mode.

The use of a negative form (no) of the command removes a created list.

Syntax

```
wan load-balance target-list <NAME>
```

```
no wan load-balance target-list { <NAME> | all }
```

Parameters

<NAME> – list name, set by the string of up to 31 characters. The value 'all' is used when deleting all lists of IP addresses to verify the integrity of the connection.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# wan load-balance target-list ten1
```

wan load-balance target-list

This command binds the list of IP addresses to verify the integrity of the connection on the network interface.

The use of a negative form (no) of the command removes a created list.

Syntax

```
wan load-balance target-list <NAME>
```

```
no wan load-balance target-list
```

Parameters

<NAME> – list name, set by the string of up to 31 characters. The value 'all' is used when deleting all lists of IP addresses to verify the integrity of the connection.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

Example

```
esr(config)# wan load-balance target-list ten1
```

wan load-balance target-list check-all

This command will be checking the IP addresses from the integrity check list. If one of the nodes being checked is unavailable, the gateway will be considered to be unavailable.

The use of a negative form (no) of the command cancells the check of the IP addresses from the integrity check list.

Syntax

`[no] wan load-balance target-list check-all`

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

Example

```
esr(config-gre)# wan load-balance target-list check-all
```

show wan rules

This command displays the operational information on the WAN rules for IPv4.

Syntax

`show wan rules <ID>`

Parameters

<ORDER> – rule number, takes values of [1..50].

Required privilege level

1

Command mode

ROOT

Example

```
esr# sh wan rules 1
Rule 1 detailed information:
    Failover:     Enabled
    Network: 11.11.11.0/24 Metric: 50
        gi1/0/1 Weight: 2 Nexthop: 3.3.3.1 [Active]
```

show wan cellular status modem

This command displays operational information about the state of interfaces with WAN enabled for the IPv4 stack.

Syntax

```
show wan cellular status modem [ <MODEM-ID> ]
```

Parameters

<MODEM-ID> – identifier of a previously created 2G/3G/4G modem (only for IPv4).

You may specify several interfaces in the command. If you do not specify interface indexes, then the statuses of all interfaces of the specified group with the WAN mode enabled for the IPv4 stack will be displayed. If a certain interface is specified, the detailed information on this interface will be displayed. If the command is executed without a parameter, the status of all system interfaces with the WAN mode enabled for the IPv4 stack will be shown.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show wan  interfaces status gigabitethernet 1/0/1
Interface 'gigabitethernet 1/0/1' detailed information:
    State:             Active
    Nexthop address:   3.3.3.1
    Last time success: 0 secs
    Last time failure: 1 hour, 48 minutes and 9 seconds
    Uptime:            --
    Targets:
        3.3.3.1 Test/Ping State: pass Attempts: 1/1
```

show wan interfaces status

This command displays operational information about the state of interfaces with WAN enabled for the IPv4 stack.

Syntax

```
show wan interfaces status [ <IF> ]
```

Parameters

<IF> – an interface, specified in the form described in Section Types and naming order of router interfaces.

You may specify several interfaces in the command. If you do not specify interface indexes, then the statuses of all interfaces of the specified group with the WAN mode enabled for the IPv4 stack will be displayed. If a certain interface is specified, the detailed information on this interface will be displayed. If the command is executed without a parameter, the status of all system interfaces with the WAN mode enabled for the IPv4 stack will be shown.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show wan  interfaces status gigabitethernet 1/0/1
Interface 'gigabitethernet 1/0/1' detailed information:
    State:             Active
    Nexthop address:   3.3.3.1
    Last time success: 0 secs
    Last time failure: 1 hour, 48 minutes and 9 seconds
    Uptime:            --
    Targets:
        3.3.3.1 Test/Ping State: pass Attempts: 1/1
```

show wan tunnels status

This command displays operational information about the state of tunnels with WAN enabled for the IPv4 stack.

Syntax

```
show wan tunnels status [ <TUN> ]
```

Parameters

<INDEX> – the identifier of the tunnel is specified as described in Section Types and naming order of router tunnels.

You may specify several tunnels in command. If you do not specify interface indexes, then the statuses of all tunnels of the specified group with the WAN mode enabled for the IPv4 stack will be displayed. If a certain tunnel is specified, the detailed information on this tunnel will be displayed. If the command is executed without a parameter, the status of all tunnels with the WAN mode enabled for the IPv4 stack will be shown.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show wan tunnels status gre 1
Tunnel 'gre 1' detailed information:
    State:              Active
    Nexthop address:    5.5.5.2
    Last time success: 0 secs
    Last time failure: 50 minutes and 2 seconds
    Uptime:             --
    Targets:
        5.5.5.2 Test/Ping State: pass Attempts: 1/1
```

show ipv6 wan rules

This command displays the operational information on the WAN rules for IPv6.

Syntax

`show ipv6 wan rules <ID>`

Parameters

<ORDER> – rule number, takes values of [1..50].

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 wan rules 1
Rule 1 detailed information:
    Failover:     Enabled
    Network: 2a14::/120 Metric: 0
        bridge 1 Weight: 1 Nexthop: 2a12::1 [Active]
```

show ipv6 wan interfaces status

This command displays operational information about the state of interfaces with WAN enabled for the IPv6 stack.

Syntax

```
show ipv6 wan interfaces status [ <IF> ]
```

Parameters

<IF> – an interface, specified in the form described in Section Types and naming order of router interfaces.

You may specify several interfaces in the command. If you do not specify interface indexes, then the statuses of all interfaces of the specified group with the WAN mode enabled for the IPv6 stack will be displayed. If a certain interface is specified, the detailed information on this interface will be displayed. If the command is executed without a parameter, the status of all system interfaces with the WAN mode enabled for the IPv6 stack will be shown.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 wan interfaces status bridge 1
Interface 'bridge 1' detailed information:
    State:             Active
    Nexthop address:   2a12::1
    Source address:    fe80::aaf9:4bff:feaa:2dcf
    Last time success: 0 secs
    Last time failure: 19 minutes and 29 seconds
    Uptime:            --
    Targets:
        2a12::1 Test/Ping State: pass Attempts: 1/1
```

26 Security

- Access control list (ACL) management
- Firewall management
- Management of logging and protection against network attacks
- Filtering management
- Intrusion Prevention System (IPS/IDS) configuration
- "Antispam" system configuration

Access control list (ACL) management

- action
- description
- enable
- ip access-list extended
- match cos
- match destination-address
- match destination-mac
- match destination-port
- match dscp
- match ip-precedence
- match protocol
- match source-address
- match source-mac
- match source-port
- match vlan
- rule
- service-acl input
- show ip access-list

action

The command specifies the action that should be applied for the traffic meeting this requirements.

The use of a negative form (no) of the command removes an assigned action.

Syntax

```
action <ACT>
```

```
no action
```

Parameters

<ACT> – allocated action:

- permit – traffic transfer is permitted;
- deny – traffic transfer is denied.

Required privilege level

10

Command mode

CONFIG-ACL-RULE

Example

```
esr(config-acl-rule)# action permit
```

description

The command is used to change a description of configured access control list.

The use of a negative form (no) of the command removes a specified description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – access control list description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-ACL

Example

```
esr(config-acl)# description "Drop SSH traffic"
```

enable

The command enables a rule.

The use of a negative form (no) of the command enables a rule.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

Rule is disabled

Required privilege level

10

Command mode

CONFIG-ACL-RULE

Example

```
esr(config-acl-rule)# enable
```

ip access-list extended

The command is used to create an access control list and to switch to the list configuration mode.

The use of a negative form (no) of the command removes a specified access control list.

Syntax

`[no] ip access-list extended <NAME>`

Parameters

<NAME> – access control list name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip access-list extended acl-ssh-drop
esr(config-acl)#
```

match cos

The command sets 802.1p priority value for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

`match cos <COS>`

`no match cos`

Parameters

<COS> – priority 802.1p value, takes values of [0..7].

Required privilege level

10

Command mode

CONFIG-ACL-RULE

Example

```
esr(config-acl-rule)# match cos 2
```

match destination-address

The command sets destination IP addresses for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match destination-address { <ADDR> <MASK> | any }
no match destination-address
```

Parameters

<ADDR> – receiver IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<MASK> – IP address mask, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. Mask bits, set to zero, specify IP address bits excluded from the comparison when searching.

When specifying the 'any' value, the rule will be triggered for any source IP address.

Default value

any

Required privilege level

10

Command mode

CONFIG-ACL-RULE

Example

```
esr(config-acl-rule)# match destination-address 10.10.10.0 255.255.255.0
```

match destination-mac

The command sets destination MAC addresses for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match destination-mac <ADDR> <WILDCARD>
```

```
no match destination-mac
```

Parameters

<ADDR> – receiver MAC address, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF];

<WILDCARD> – MAC address mask, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF]. Mask bits, set to zero, specify MAC address bits excluded from the comparison when searching.

Required privilege level

10

Command mode

CONFIG-ACL-RULE

Example

```
esr(config-acl-rule)# match destination-mac A8:F9:4B:AA:00:41 00:00:00:00:00:FF
```

match destination-port

The command sets the number of source TCP/UDP port for which the rule should work.

The use of a negative form (no) of the command removes the assignment.

Syntax

```
match destination-port {<PORT> | any}
```

```
no match destination-port
```

Parameters

<PORT> – number of destination TCP/UDP port, takes values of [1..65535]. When specifying the 'any' value, the rule will be triggered for any source TCP/UDP port.

Default value

any

Required privilege level

10

Command mode

CONFIG-ACL-RULE

Example

```
esr(config-acl-rule)# match destination-port 22
```

match dscp

The command sets DSCP code value for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match dscp <DSCP>
no match dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Required privilege level

10

Command mode

CONFIG-ACL-RULE

Example

```
esr(config-acl-rule)# match dscp 55
```

match ip-precedence

The command sets IP Precedence code value for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match ip-precedence <IPP>
no match ip-precedence
```

Parameters

<IPP> – IP Precedence code value, takes values in the range of [0..7].

Required privilege level

10

Command mode

CONFIG-ACL-RULE

Example

```
esr(config-acl-rule)# match ip-precedence 5
```

match protocol

The command sets name of IP for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match protocol <TYPE>
no match protocol
match protocol-id <ID>
no match protocol-id
```

Parameters

<TYPE> – protocol type, takes the following values: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre. When specifying the 'any' value, the rule will work for any protocols.

<ID> – IP identification number, takes values of [0x00-0xFF].

When specifying the 'any' value, the rule will work for any protocols.

Default value

any

Required privilege level

10

Command mode

CONFIG-ACL-RULE

Example

```
esr(config-acl-rule)# match protocol tcp
```

match source-address

The command sets source IP addresses for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match source-address { <ADDR> <MASK> | any }
no match source-address
```

Parameters

<ADDR> – source IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<MASK> – IP address mask, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. Mask bits, set to zero, specify IP address bits excluded from the comparison when searching.

When specifying the 'any' value, the rule will be triggered for any source IP address.

Default value

any

Required privilege level

10

Command mode

CONFIG-ACL-RULE

Example

```
esr(config-acl-rule)# match source-address 10.100.100.0 255.255.255.0
```

match source-mac

The command sets source MAC addresses for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match source-mac <ADDR> <WILDCARD>
no match source-mac
```

Parameters

<ADDR> – source MAC address, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF].

<WILDCARD> – MAC address mask, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF]. Mask bits, set to zero, specify MAC address bits excluded from the comparison when searching.

Required privilege level

10

Command mode

CONFIG-ACL-RULE

Example

```
esr(config-acl-rule)# match source-mac A8:F9:4B:AA:00:40 00:00:00:FF:FF:FF
```

match source-port

The command sets the number of source TCP/UDP port for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match source-port { <PORT> | any }
no match source-port
```

Parameters

<PORT> – number of source TCP/UDP port, takes values of [1..65535]. When specifying the 'any' value, the rule will be triggered for any source TCP/UDP port.

Default value

any

Required privilege level

10

Command mode

CONFIG-ACL-RULE

Example

```
esr(config-acl-rule)# match source-port any
```

match vlan

The command sets VLAN ID value for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match vlan <VID>
no match vlan
```

Parameters

<VID> – VLAN ID, takes values of [1..4094].

Required privilege level

10

Command mode

CONFIG-ACL-RULE

Example

```
esr(config-acl-rule)# match vlan 100
```

rule

The command creates a rule and switches to CONFIG-ACL-RULE configuration mode. The rules are proceeded by the device in number ascending order.

The use of a negative form (no) of the command removes a specified rule.

Syntax

```
[no] rule <ORDER>
```

Parameters

<ORDER> – rule number, takes values of [1..4094].

Required privilege level

10

Command mode

CONFIG-ACL

Example

```
esr(config-acl)# rule 10
esr(config-acl-rule)#
```

service-acl input

The command binds a specified access control list to a configured interface to filtrate incoming traffic.

The use of a negative form (no) of the command removes the binding of access control list to this interface.

Syntax

```
service-acl input <NAME>
no service-acl input
```

Parameters

<NAME> – access control list name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-PORT-CHANNEL

Example

```
esr(config-if-gi)# service-acl input acl-ssh-drop
```

show ip access-list

The command displays access control lists.

Syntax

```
show ip access-list [ <NAME> [ <ORDER> ] ]
```

Parameters

<NAME> – access control list name, set by the string of up to 31 characters;

<ORDER> – rule number, takes values of [1..4096]. When specifying a rule number, only the given rules information will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip access-list
Name                               Description
--------------------------------   -----------------------------------------------
acl-telnet-drop                    --
acl-ssh-drop                       Drop SSH traffic
esr# show ip access-list acl-ssh-drop
Index:                        1
Matching pattern:
    Protocol:                 TCP(6)
    Source MAC address:       any
    Source IP address:        any
    Source port:              any
    Destination MAC address:  any
    Destination IP address:   any
    Destination port:         22
Action:                       Deny
Status:                       Enabled
--------------------------------------------------------------------------------
Index:                        2
Matching pattern:
    Protocol:                 any
    Source MAC address:       any
    Source IP address:        any
    Destination MAC address:  any
    Destination IP address:   any
Action:                       Permit
Status:                       Enabled
--------------------------------------------------------------------------------
```

Firewall management

- action
- clear ip firewall counters
- clear ip firewall sessions
- clear ipv6 firewall counters
- clear ipv6 firewall sessions
- description
- enable
- ip firewall disable
- ip firewall mode
- ip firewall sessions counters
- ip firewall sessions allow-unknown
- ip firewall sessions generic-timeout
- ip firewall sessions icmp-timeout
- ip firewall sessions icmpv6-timeout
- ip firewall sessions max-expect
- ip firewall sessions max-tracking
- ip firewall sessions tcp-connect-timeout
- ip firewall sessions tcp-disconnect-timeout
- ip firewall sessions tcp-estabilished-timeout
- ip firewall sessions tcp-latecome-timeout
- ip firewall sessions tracking
- ip firewall sessions udp-assured-timeout
- ip firewall sessions udp-wait-timeout
- match application
- match destination-address
- match destination-address-port
- match destination-mac
- match destination-nat
- match destination-port
- match fragment
- match icmp
- match ip-option
- match protocol
- match source-address
- match source-address-port
- match source-mac
- match source-port
- ports firewall enable
- rate-limit pps
- rearrange
- renumber rule
- rule
- security zone
- security-zone
- security zone-pair
- show ip firewall counters
- show ip firewall sessions
- show ip firewall sessions tracking
- show ipv6 firewall counters
- show ipv6 firewall sessions
- show security zone

- show security zone-pair
- show security zone-pair configuration

action

The command specifies the action that should be applied for the traffic meeting this requirements.

The use of a negative form (no) of the command removes an assigned action.

Syntax

```
action <ACT> [log]

no action
```

Parameters

<ACT> – allocated action:

- permit – traffic transfer is permitted;
- deny – traffic transfer is denied.
- reject – traffic passing is prohibited, error notification is sent to user;
- netflow-sample – traffic passing is allowed, statistics export via Netflow is performed;
- sflow-sample – traffic passing is allowed, statistics export via sFlow is performed;
- log – activation key for logging of sessions established according to this rule.

Default value

Action is not configured, logging is disabled.

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# action permit
```

clear ip firewall counters

The command resets Firewall rule counters.

Syntax

```
clear ip firewall counters [ vrf <VRF> ] [ <SOURCE-ZONE> [ <DESTINATION-ZONE>
[<ORDER>] ] ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, rule counters in a specified VRF will be cleared;

<SOURCE-ZONE> – security zone from which traffic flows;

<DESTINATION-ZONE> – security zone to which traffic flows;

<ORDER> – rule number, takes values of [1..10000]. When specifying a rule number, only the given rule's counters will be cleared.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear ip firewall counters trusted self
```

clear ip firewall sessions

The command removes active IP sessions.

Syntax

```
clear ip firewall sessions [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source-address
<ADDR> ] [ outiside-source-address <ADDR> ] [ inside-destination-address <ADDR> ]
[ outside-destination-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-source-
port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port <PORT> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, active sessions in a specified VRF will be removed;

<TYPE> – protocol type, takes the following values: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<PORT> – TCP/UDP port, takes values of [1..65535];

inside-source-address – key to specify source IP address of incoming packets;

inside-destination-address – key to specify destination IP address of incoming packets;

outside-source-address – key to specify source IP address of outgoing packets;

outside-destination-address – key to specify destination IP address of outgoing packets;

inside-source-port – key to specify sender TCP/UDP port of incoming packets;

outside-source-port – key to specify sender TCP/UDP port of outgoing packets;

inside-destination-port – key to specify receiver TCP/UDP port of incoming packets;

outside-destination-port – key to specify receiver TCP/UDP port of outgoing packets;

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear ip firewall sessions vrf VRF1
```

clear ipv6 firewall counters

The command resets Firewall rule counters.

Syntax

```
clear ipv6 firewall counters [ vrf <VRF> ] [ <SOURCE-ZONE> [ <DESTINATION-ZONE>
[<ORDER>] ] ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, rule counters in a specified VRF will be cleared.

<SOURCE-ZONE> – security zone from which traffic flows.

<DESTINATION-ZONE> – security zone to which traffic flows.

<ORDER> – rule number, takes values of [1..10000]. When specifying a rule number, only the given rule's counters will be cleared.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear ipv6 firewall counters trusted self
```

clear ipv6 firewall sessions

The command removes active IPv6 sessions.

Syntax

```
clear ipv6 firewall sessions [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source-address
<IPV6-ADDR> ] [ outiside-source-address <IPV6-ADDR> ] [ inside-destination-address <IPV6-
ADDR> ] [ outside-destination-address <IPV6-ADDR> ] [ inside-source-port <PORT> ]
[ outside-source-port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-
port <PORT> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, active sessions in a specified VRF will be removed.

<TYPE> – protocol type, takes the following values: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre.

<IPV6-ADDR> – IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

<PORT> – TCP/UDP port, takes values of [1..65535];

inside-source-address – command to specify IPv6 source address of incoming packets.

inside-destination-address – command to specify IPv6 destination address of incoming packets.

outiside-source-address – command to specify IPv6 source address of outgoing packets;

outside-destination-address – command to specify IPv6 destination address of outgoing packets;

inside-source-port – key to specify sender TCP/UDP port of incoming packets;

outside-source-port – key to specify sender TCP/UDP port of outgoing packets;

inside-destination-port – key to specify receiver TCP/UDP port of incoming packets;

outside-destination-port – key to specify receiver TCP/UDP port of outgoing packets;

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear ipv6 firewall sessions vrf VRF1
```

description

The command changes the description of configured zone or a pair of security zones. The use of a negative form (no) of the command removes a specified description.

Syntax

```
description <DESCRIPTION>
```

```
no description
```

Parameters

<DESCRIPTION> – security zone name, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-ZONE

CONFIG-ZONE-PAIR

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone)# description "Trusted interfaces"
```

enable

The command enables a rule.

The use of a negative form (no) of the command enables a rule.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# enable
```

ip firewall disable

The command disables Firewall function on a network interface.

The use of a negative form (no) of the command enables Firewall function on a network interface.

Syntax

```
[no] ip firewall disable
```

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

CONFIG-MULTILINK

CONFIG-CELLULAR-MODEM

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-L2TP

CONFIG-LT

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-OPENVPN

Example

```
esr(config-if-gi)# ip firewall disable
```

ip firewall mode

The command selects firewall operation mode.

The use of a negative form (no) of the command sets the default firewall operation mode.

Syntax

```
ip firewall mode <MODE>
no ip firewall mode
```

Parameters

<MODE> – firewall operation mode, may take the following values:

- stateful – mode where the router monitors sessions. The first session packets are undergone a full verification cycle according to firewall rules and the following session packets are routed without additional verifications. This rule is not valid for DPI mechanism operation.
- stateful – mode where the router does not monitor sessions. Each packet is undergone a full verification cycle according to firewall rules that significantly reduces the equipment performance. The use of this mode is allowed only when strictly necessary.

Default value

stateful

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config-if-gi)# ip firewall mode stateless
```

ip firewall sessions counters

The command enables session counters for NAT and Firewall. The counters increase only when a new session is established. For established sessions, increase of counters value does not occur during packets transmission. Enabling the counters reduces the router performance.

The commands for viewing counters and sessions are described in sections show ip firewall counters, show ip firewall sessions, show ipv6 firewall counters and show ipv6 firewall sessions.

The use of a negative form (no) of the command disables session counters.

Syntax

```
[no] ip firewall sessions counters
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions counters
```

ip firewall sessions allow-unknown

The command disables filtration of packets for which it was not possible to define belonging to any known connection and which are not the beginning of a new connection.

The use of a negative form (no) of the command enables the interface being configured.

Syntax

```
[no] ip firewall sessions allow-unknown
```

Parameters

The command does not contain parameters.

Default value

Enabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions allow-unknown
```

ip firewall sessions generic-timeout

The command specifies session lifetime for unsupported protocols after which it is considered to be outdated and is removed from the trackable session table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip firewall sessions generic-timeout <TIME>
no ip firewall sessions generic-timeout
```

Parameters

<TIME> – session lifetime for unsupported protocols, takes values in seconds [1..8553600].

Default value

60 seconds

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions generic-timeout 60
```

ip firewall sessions icmp-timeout

The command specifies ICMP session lifetime after which it is considered to be outdated and is removed from the trackable session table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip firewall sessions icmp-timeout <TIME>
no ip firewall sessions icmp-timeout
```

Parameters

<TIME> – ICMP session lifetime, takes values in seconds [1..8553600].

Default value

30 seconds

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions icmp-timeout 60
```

ip firewall sessions icmpv6-timeout

The command specifies ICMPv6 session lifetime after which it is considered to be outdated and is removed from the trackable session table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip firewall sessions icmpv6-timeout <TIME>
no ip firewall sessions icmpv6-timeout
```

Parameters

<TIME> – ICMPv6 session lifetime, takes values in seconds [1..8553600].

Default value

30 seconds

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions icmpv6-timeout 60
```

ip firewall sessions max-expect

The command defines the size of outstanding session table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip firewall sessions max-expect <COUNT>
no ip firewall sessions max-expect
```

Parameters

<COUNT> – table size, takes values of [1..8553600].

Default value

256

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions max-expect 512
```

ip firewall sessions max-tracking

The command defines the size of trackable session table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip firewall sessions max-tracking <COUNT>
no ip firewall sessions max- tracking
```

Parameters

<COUNT> – table size, takes values of [1..8553600].

Default value

512000

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions max-tracking 256000
```

ip firewall sessions tcp-connect-timeout

The command defines the lifetime of TCP session in 'connection is being established' state after which it is considered to be outdated and is removed from trackable session table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip firewall sessions tcp-connect-timeout <TIME>
no ip firewall sessions tcp-connect-timeout
```

Parameters

<TIME> – lifetime of TCP session in 'connection is being established' state, takes values in seconds [1..8553600].

Default value

60 seconds

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions tcp-connect-timeout 120
```

ip firewall sessions tcp-disconnect-timeout

The command defines the lifetime of TCP session in 'connection is closed' state after which it is considered to be outdated and is removed from trackable session table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip firewall sessions tcp-disconnect-timeout <TIME>
```

```
no ip firewall sessions tcp-disconnect-timeout
```

Parameters

<TIME> – lifetime of TCP session in 'connection is being closed' state, takes values in seconds [1..8553600].

Default value

30 seconds

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions tcp-disconnect-timeout 10
```

ip firewall sessions tcp-estabilished-timeout

The command defines the lifetime of TCP session in 'connection is established' state after which it is considered to be outdated and is removed from trackable session table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip firewall sessions tcp-estabilished-timeout <TIME>
```

```
no ip firewall sessions tcp-estabilished-timeout
```

Parameters

<TIME> – lifetime of TCP session in 'connection is being established' state, takes values in seconds [1..8553600].

Default value

120 seconds

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions tcp-estabilished-timeout 3600
```

ip firewall sessions tcp-latecome-timeout

The command defines the timeout after which the closed TCP session is actually deleted from the table of trackable sessions.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip firewall sessions tcp-latecome-timeout <TIME>
no ip firewall sessions tcp-latecome-timeout
```

Parameters

<TIME> – timeout, takes value in seconds [1..8553600].

Default value

120 seconds

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions tcp-latecome-timeout 10
```

ip firewall sessions tracking

The command enables the function of application level session tracking for certain protocols.

The use of a negative form (no) of the command disables the function of application level session tracking for certain protocols.

Syntax

```
ip firewall sessions tracking { <PROTOCOL> |  sip [ port <OBJECT-GROUP-SERVICE> ] }
no ip firewall sessions tracking { <PROTOCOL> |  sip [ port <OBJECT-GROUP-SERVICE> ] |
all }
```

Parameters

<PROTOCOL> – application level protocol, sessions of which should be monitored, takes the values of [ftp, h323, pptp, netbios-ns].

<OBJECT-GROUP-SERVICE> – sip session TCP/UDP ports' profile name, set by the string of up to 31 characters. If a group is not specified, sip sessions monitoring will be performed for 5060 port.

Instead of a certain protocol you can use the 'all' key that enables application-level session tracking for all available protocols.

Default value

Disabled for all protocols.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions tracking ftp
```

ip firewall sessions udp-assured-timeout

The command defines the lifetime of UDP session in 'connection is confirmed' state after which it is considered to be outdated and is removed from trackable session table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip firewall sessions udp-assured-timeout <TIME>

no ip firewall sessions udp-assured-timeout
```

Parameters

<TIME> – lifetime of UDP session in 'connection is confirmed' state, takes values in seconds [1..8553600].

Default value

180 seconds

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions udp-assured-timeout 3600
```

ip firewall sessions udp-wait-timeout

The command defines the lifetime of UDP session in 'connection is not confirmed' state after which it is considered to be outdated and is removed from trackable session table.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip firewall sessions udp-wait-timeout <TIME>
no ip firewall sessions udp-wait-timeout
```

Parameters

<TIME> – lifetime of UDP session in 'connection is not confirmed' state, takes values in seconds [1..8553600].

Default value

30 seconds

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions udp-wait-timeout 60
```

match application

The command sets application profile for which the rule should work. The function is used for filtration on the basis of applications (DPI mechanism)

When using 'not' parameter, the rule will work for applications which are not included in a specified profile.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] application <OBJ-GROUP-APPLICATION>
```

```
no match application
```

Parameters

<OBJ-GROUP-APPLICATION> – application profile name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# match application APP_DENY
```

match destination-address

The command sets the profile of destination IP addresses for which the rule should work.

When using 'not' parameter, the rule will work for destination IP addresses which are not included in a specified profile.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] destination-address <OBJ-GROUP-NETWORK-NAME>
```

```
no match destination-address
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – IP addresses profile name, set by the string of up to 31 characters. When specifying the 'any' value, the rule will be triggered for any source IP address.

Default value

any

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# match destination-address local
```

match destination-address-port

This command sets the profile of IP address bundles and destination TCP/UDP ports for which the rule should work.

When using 'not' parameter, the rule will work for IP address bundles and destination TCP/UDP ports which are not included in a specified profile.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] destination-address-port <OBJ-GROUP-ADDRESS-PORT-NAME>
no match destination-address
```

Parameters

<OBJ-GROUP-ADDRESS-PORT-NAME> – the name of the profile of IP address bundles and TCP/UDP ports is specified by a string of up to 31 characters. When specifying the value 'any', the rule will not consider this filtering method.

Default value

any

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# match destination-address local
```

match destination-mac

The command sets destination MAC address for which the rule should work.

When using 'not' (match not) parameter, the rule will work for destination MAC addresses different from a specified one.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] destination-mac <ADDR>
no match destination-mac <ADDR>
```

Parameters

<ADDR> – destination MAC address, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF].

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# match destination-mac A8:F9:4B:AA:00:40
```

match destination-nat

The command sets the limitation under which the rule will only work for traffic modified by the IP address and destination ports translation service.

When using 'not' parameter, the rule will work for traffic not modified by the IP address and destination ports translation service. The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] destination-nat
no match destination-nat
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# match destination-nat
```

match destination-port

This command sets the profile of destination TCP/UDP ports for which the rule should work.

When using 'not' parameter, the rule will work for destination TCP/UDP ports which are not included in a specified profile.

The use of a negative form (no) of the command removes the assignment.

Syntax

```
match [not] destination-port <PORT-SET-NAME>
no match destination-port
```

Parameters

<PORT-SET-NAME> – TCP/UDP ports profile name, set by the string of up to 31 characters. When specifying the 'any' value, the rule will be triggered for any source TCP/UDP port.

Default value

any

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# match destination-port ssh
```

match fragment

The command defines fragmented packets sent to the device. The command is applicable only in rules between ‘any self’ zones. The second and the following fragments of a packet are subject to the rule. Packets are proceeded by the rule before DNAT addresses translation.

When using 'not' parameter, the rule will work for non fragmented packets.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] fragment
no match fragmen
```

Parameters

None.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-pair-rule)# match fragment
```

match icmp

The command is used to configure ICMP parameters if it is selected by 'match protocol' command. The command specifies the type and code of ICMP messages for which the rule should work.

When using 'not' parameter, the rule will work for all types and codes of ICMP messages excluding specified ones.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] icmp { <ICMP_TYPE> <ICMP_CODE> | <OPTION> }
```

```
no match icmp
```

Parameters

<ICMP_TYPE> – ICMP message type, takes values of [0..255].

<ICMP_CODE> – ICMP message code, takes values of [0..255]. When specifying the “any” value, the rule will work for any ICMP message code

<OPTION> – standard types of ICMP messages can take values:

- administratively-prohibited;
- alternate-address;
- conversion-error;
- dod-host-prohibited;
- dod-network-prohibited;
- echo;
- echo-reply;
- host-isolated;
- host-precedence;
- host-redirect;
- host-tos-redirect;
- host-tos-unreachable;
- host-unknown;
- host-unreachable;
- information-reply;
- information-request;
- mask-reply;
- mask-request;
- network-redirect;
- network-tos-redirect;
- network-tos-unreachable;
- network-unknown;
- network-unreachable;
- option-missing;
- packet-too-big;
- parameter-problem;
- port-unreachable;
- precedence;
- protocol-unreachable;
- reassembly-timeout;
- router-advertisement;
- router-solicitation;
- source-quench;
- source-route-failed;
- time-exceeded;
- timestamp-reply;
- timestamp-request;
- traceroute.

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# match icmp 2 any
```

match ip-option

The command defines packets which contain options in IP headers. The command is applicable only in rules between ‘any self’ zones.

When using 'not' parameter, the rule will work for packets which do not contain options in IP headers.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] ip-option
no match ip-option
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-pair-rule)# match ip-options
```

match protocol

The command sets name or number of IP for which the rule should work.

When using 'not' parameter, the rule will work for all protocols except a specified one.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] protocol <TYPE>
no match protocol
match [not] protocol-id <ID>
no match protocol-id
```

Parameters

<TYPE> – protocol type, takes the following values: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre.

When specifying the 'any' value, the rule will work for any protocols.

<ID> – IP identification number, takes values of [0x00-0xFF].

Default value

any

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# match protocol udp
```

match source-address

The command specifies the profile of source IP addresses for which the rule should work.

When using 'not' (match not) parameter, the rule will work for source IP addresses which are not included in a specified profile.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] source-address <OBJ-GROUP-NETWORK-NAME>
no match source-address <OBJ-GROUP-NETWORK-NAME>
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – IP addresses profile name, set by the string of up to 31 characters. When specifying the 'any' value, the rule will be triggered for any source IP address.

Default value

any

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# match source-address remote
```

match source-address-port

This command sets the profile of IP address bundles and source TCP/UDP ports for which the rule should work.

When using 'not' (match not) parameter, the rule will work for IP address bundles and source TCP/UDP ports which are not included in a specified profile.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] source-address-port <OBJ-GROUP-ADDRESS-PORT-NAME>
no match source-address-port <OBJ-GROUP-ADDRESS-PORT-NAME>
```

Parameters

<OBJ-GROUP-ADDRESS-PORT-NAME> – the name of the profile of IP address bundles and TCP/UDP ports is specified by a string of up to 31 characters. When specifying the value 'any', the rule will not consider this filtering method.

Default value

any

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# match source-address-port admin
```

match source-mac

The command sets source MAC address for which the rule should work.

When using 'not' (match not) parameter, the rule will work for source MAC addresses different from a specified one.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] source-mac <ADDR>
no match source-mac <ADDR>
```

Parameters

<ADDR> – source MAC address, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF].

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# match source-mac A8:F9:4B:AA:00:40
```

match source-port

The command sets the profile of source TCP/UDP ports for which the rule should work.

When using 'not' parameter, the rule will work for source TCP/UDP ports which are not included in a specified profile.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
match [not] source-port <PORT-SET-NAME>

no match source-port
```

Parameters

<PORT-SET-NAME> – TCP/UDP ports profile name, set by the string of up to 31 characters. When specifying the 'any' value, the rule will be triggered for any source TCP/UDP port.

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-zone-rule)# match source-port telnet
```

ports firewall enable

The command enables session filtration and monitoring during packets transmission between Bridge interface members.

The use of a negative form (no) of the command removes an assigned action.

Syntax

```
[no] ports firewall enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-BRIDGE

Example

```
esr(config-bridge)# ports firewall enable
```

rate-limit pps

The command limits the amount of packets per second proceeded by the rule. The command can be used only in rules between any self zones and subject to the action permit action in this rule.

The use of a negative form (no) of the command removes an assigned action.

Syntax

`rate-limit pps <RATE>`

`no rate-limit`

Parameters

<PPS> – amount of MAC addresses per second, takes value of [1..10000].

Default value

Not limited.

Required privilege level

15

Command mode

CONFIG-ZONE-PAIR-RULE

Example

```
esr(config-if-gi)# rate-limit pps 200
```

rearrange

This command changes the step between the created rules.

Syntax

`rearrange <VALUE>`

Parameters

<VALUE> – maximum between rules, takes values of [1..50].

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR

Example

```
esr(config-zone-pair)# rearrange 10
```

renumber rule

This command changes the rule number.

Syntax

```
renumber rule <CUR_ORDER> <NEW_ORDER>
```

Parameters

<CUR_ORDER> – current rule number, takes values of [1..10000].

<NEW_ORDER> – new rule number, takes values of [1..10000].

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR

Example

```
esr(config-zone-pair)# renumber rule 13 100
```

rule

The command is used to create the rule and to switch to SECURITY ZONE PAIR RULE command mode. The rules are proceeded by the device in number ascending order.

The use of a negative form (no) of the command removes a specified rule.

Syntax

```
[no] rule <ORDER>
```

Parameters

<ORDER> – rule number, takes values of [1..10000]. If the «all» parameter value is used when removing, all rules for the configured security zone pairs will be removed.

Required privilege level

10

Command mode

CONFIG-ZONE-PAIR

Example

```
esr(config-zone-pair)# rule 10
esr(config-zone-rule)#
```

security zone

The command is used to create a security zone and to switch to the zone edit mode.

The use of a negative form (no) of the command removes a specified security zone.

Syntax

```
[no] security zone [ <NAME> | all ]
```

Parameters

<NAME> – security zone name, set by the string of up to 12 characters. The use of a negative form (no) of the command with ‘all’ parameter removes all security zones.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# security zone trusted
esr(config-zone)#
```

security-zone

The command adds a current network interface to security zone. The use of a negative form (no) of the command removes interface from the zone.

Syntax

```
security-zone <NAME>
```

```
no security-zone
```

Parameters

<NAME> – security zone name, set by the string of up to 12 characters.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-CELLULAR-MODEM

CONFIG-E1

CONFIG-MULTILINK

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-L2TP-SERVER

CONFIG-OPENVPN-SERVER

CONFIG-PPTP-SERVER

Example

```
esr(config-if-gi)# security-zone trusted
```

security zone-pair

The command creates rule group for a pair of security zones.

The use of a negative form (no) of the command removes a specified rule group.

Syntax

```
[no] security zone-pair <SOURCE-ZONE> <DESTINATION-ZONE>
```

Parameters

<SOURCE-ZONE> – security zone from which traffic flows;

<DESTINATION-ZONE> – security zone to which traffic flows. Router always has a security zone named 'self'. When the traffic recipient is the router itself, i.e. traffic is not transit, pass 'self' zone as a parameter. If the 'all' parameter value is used when removing, all configurable pairs of security zones will be removed.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# security zone-pair trusted self
```

show ip firewall counters

This command dispays statistics on packets that pass between zones for which no session is established.

Syntax

```
show ip firewall counters [ vrf <VRF> ] [ <SOURCE-ZONE> [ <DESTINATION-ZONE>
[ <ORDER> ] ] ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, rule counters in a specified VRF will be shown;

<DESTINATION-ZONE> – security zone to which traffic flows;

<SOURCE-ZONE> – security zone from which traffic flows;

<ORDER> – rule number, takes values of [1..10000]. When specifying a rule number, only the given rules information will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip firewall counters trusted self
Zone-pair                         Rule         Action             Pkts         Bytes
------------------------------    ----------   ---------------    ----------   ----------
any/any                           default      deny               0            0
trusted/self                      1            permit             0            0
trusted/trusted                   1            permit             0            0
```

show ip firewall sessions

The command displays active IP sessions.

Syntax

```
show ip firewall sessions [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source-address
<ADDR>] [ outside-source-address <ADDR> ] [ inside-destination-address <ADDR> ]
[ outside-destination-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-source-
port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port <PORT> ]
[ summary ] [ configuration ] [ expected ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, active sessions will be displayed in a specified VRF.

<TYPE> – protocol type, takes the following values: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<PORT> – TCP/UDP port, takes values of [1..65535];

inside-source-address – command to specify IP source address of incoming packets;

inside-destination-address – command to specify IP destination address of incoming packets;

outiside-source-address – command to specify IP source address of outgoing packets;

outside-destination-address – command to specify IP destination address of outgoing packets;

inside-source-port – key to specify sender TCP/UDP port of incoming packets;

outside-source-port – key to specify sender TCP/UDP port of outgoing packets;

inside-destination-port – key to specify receiver TCP/UDP port of incoming packets;

outside-destination-port – key to specify receiver TCP/UDP port of outgoing packets;

summary – displays summary statistics for IP sessions;

configuration – displays IP sessions timeout configuration and table volume;

expected - command to display sessions waiting to be processed by other sessions.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip firewall sessions
Prot Inside source  Inside destination  Outside source Outside destination Pkts Bytes
---  ------------  ----------------    -------------   ----------------  -----  ----
vrrp   4.4.4.4      224.0.0.18              4.4.4.4       224.0.0.18       --     --
```

show ip firewall sessions tracking

This command displays the configuration of the application session tracking functionality.

Syntax

`show ip firewall sessions tracking`

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip firewall sessions tracking
Tracking Status:
    FTP:        Enabled
    H.323:      Enabled
    GRE:        Enabled
    PPTP:       Enabled
    NETBIOS-NS: Enabled
    SIP:        Enabled
```

show ipv6 firewall counters

This command dispays statistics on packets that pass between zones for which no session is established.

Syntax

```
show ipv6 firewall counters [ vrf <VRF> ] [ <SOURCE-ZONE> [ <DESTINATION-ZONE>
[ <ORDER> ] ] ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, rule counters in a specified VRF will be shown;

<DESTINATION-ZONE> – security zone to which traffic flows;

<SOURCE-ZONE> – security zone from which traffic flows;

<ORDER> – rule number, takes values of [1..10000]. When specifying a rule number, only the given rules information will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 firewall counters trusted self
Zone-pair                        Rule         Action             Pkts         Bytes
------------------------------   ----------   ---------------    ----------   ----------
any/any                          default      deny               0            0
trusted/self                     1            permit             0            0
trusted/trusted                  1            permit             0            0
```

show ipv6 firewall sessions

The command displays active IPv6 sessions.

Syntax

```
show ipv6 firewall sessions [ vrf <VRF> ] [summary] [ protocol <TYPE> ] [ inside-source-address <IPV6-ADDR>] [ outiside-source-address <IPV6-ADDR> ] [ inside-destination-address <IPV6-ADDR> ] [ outside-destination-address <IPV6-ADDR> ] [ inside-source-port <PORT> ] [ outside-source-port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port <PORT> ] [ expected ] [ summary ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, active sessions will be displayed in a specified VRF.

<TYPE> – protocol type, takes the following values: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<IPV6-ADDR> – IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<PORT> – TCP/UDP port, takes values of [1..65535];

inside-source-address – command to specify IPv6 source address of incoming packets;

inside-destination-address – command to specify IPv6 destination address of incoming packets;

outiside-source-address – command to specify IPv6 source address of outgoing packets;

outside-destination-address – command to specify IPv6 destination address of outgoing packets;

inside-source-port – key to specify sender TCP/UDP port of incoming packets;

outside-source-port – key to specify sender TCP/UDP port of outgoing packets;

inside-destination-port – key to specify receiver TCP/UDP port of incoming packets;

outside-destination-port – key to specify receiver TCP/UDP port of outgoing packets;

expected - command to display sessions waiting to be processed by other sessions;

summary – displays summary statistics for IPv6 sessions.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 firewall sessions
Prot  Inside source  Inside destination  Outside source Outside destination  Pkts  Bytes
----- -------------- ------------------- -------------- -------------------- ----- -----
icmp6 fc00::2         fc00::2            fc00::2        fc00::2               --     --
icmp6 fc00::2         fc00::1            fc00::2        fc00::1               --     --
```

show security zone

This command displays the interfaces included in the security zone.

Syntax

```
show security zone [<NAME>]
```

Parameters

<NAME> – zone name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show security zone
Zone name       Interfaces
-------------   ------------------------------------------
trusted         gi1/0/2-6, gi1/0/8-24, bridge 1
untrusted       gi1/0/1, te1/0/1-2, bridge 2
```

show security zone-pair

The command displays zone pairs lists.

Syntax

```
show security zone-pair
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show security zone-pair
From zone       To zone
-------------   -------------
trusted         untrusted
trusted         trusted
trusted         self
untrusted       self
```

show security zone-pair configuration

The command displays rules for security zones pair.

Syntax

```
show security zone-pair configuration <SOURCE-ZONE> <DESTINATION-ZONE> [<ORDER>]
```

Parameters

<SOURCE-ZONE> – security zone from which traffic flows;

<DESTINATION-ZONE> – security zone to which traffic flows;

<ORDER> – rule number, takes values of [1..10000]. When specifying a rule number, only the given rules information will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show security zone-pair configuration trusted self
Order:             1
Description:       --
Matching pattern:
    Protocol:      tcp(6)
    Src-addr:      any
    src-port:      any
    Dest-addr:     any
    dest-port:     23
0            0
```

Management of logging and protection against network attacks

- Management of protection against network attacks
  - ip firewall screen dos-defense icmp-threshold
  - ip firewall screen dos-defense land
  - ip firewall screen dos-defense limit-session-destination
  - ip firewall screen dos-defense limit-session-source
  - ip firewall screen dos-defense syn-flood
  - ip firewall screen dos-defense udp-threshold
  - ip firewall screen dos-defense winnuke
  - ip firewall screen spy-blocking fin-no-ack
  - ip firewall screen spy-blocking icmp-type destination-unreachable
  - ip firewall screen spy-blocking icmp-type echo-request
  - ip firewall screen spy-blocking icmp-type reserved
  - ip firewall screen spy-blocking icmp-type source-quench
  - ip firewall screen spy-blocking icmp-type time-exceeded
  - ip firewall screen spy-blocking ip-sweep
  - ip firewall screen spy-blocking port-scan
  - ip firewall screen spy-blocking spoofing
  - ip firewall screen spy-blocking syn-fin
  - ip firewall screen spy-blocking tcp-all-flags
  - ip firewall screen spy-blocking tcp-no-flag
  - ip firewall screen suspicious-packets icmp-fragment
  - ip firewall screen suspicious-packets ip-fragment
  - ip firewall screen suspicious-packets large-icmp
  - ip firewall screen suspicious-packets syn-fragment
  - ip firewall screen suspicious-packets udp-fragment
  - ip firewall screen suspicious-packets unknown-protocols
- Management of network attacks notification
  - ip firewall logging interval
  - logging firewall screen detailed
  - logging firewall screen dos-defense
  - logging firewall screen spy-blocking
  - logging firewall screen suspicious-packets
  - show ip firewall screens counters

Management of protection against network attacks

ip firewall screen dos-defense icmp-threshold

This command enables the protection against ICMP flood attacks. When the protection is enabled, the amount of all types ICMP packets per second for one destination address is limited.

The use of a negative form (no) of the command disables ICMP flood protection.

Syntax

```
ip firewall screen dos-defense icmp-threshold { <NUM> }
```

```
no ip firewall screen dos-defense icmp-threshold
```

Parameters

<NUM> – amount of ICMP packets per second, set in the range of [1..10000]

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen dos-defense icmp-threshold 2000
```

ip firewall screen dos-defense land

This command enables the protection against land attacks. When the protection is enabled, the packets with the same source and destination IP addresses and with SYN flag in TCP header are blocked.

The use of a negative form (no) of the command disables land attacks protection.

Syntax

`[no] ip firewall screen dos-defense land`

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen dos-defense land
```

ip firewall screen dos-defense limit-session-destination

When the host IP sessions table is overfilled, the host is unable to establish new sessions and it drops the requests (this may happen during various DoS attacks: SYN flood, UDP flood, ICMP flood, etc.). The command enables limiting the number of packets transmitted per second per destination address, which attenuates DoS attacks.

The use of a negative form (no) of the command removes the restriction.

Syntax

```
ip firewall screen dos-defense limit-session-destination { <NUM> }
no ip firewall screen dos-defense limit-session-destination
```

Parameters

<NUM> – limit number of ip packets per second, set in the range of [1..10000].

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen dos-defense limit-session-destination 1000
```

ip firewall screen dos-defense limit-session-source

When the host IP sessions table is overfilled, the host is unable to establish new sessions and it drops the requests (this may happen during various DoS attacks: SYN flood, UDP flood, ICMP flood, etc.). The command enables limiting the number of packets transmitted per second per source address, which attenuates DoS attacks.

The use of a negative form (no) of the command removes the restriction.

Syntax

```
ip firewall screen dos-defense limit-session-source { <NUM> }
no ip firewall screen dos-defense limit-session-source
```

Parameters

<NUM> – limit number of ip packets per second, set in the range of [1..10000].

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen dos-defense limit-session-source 1000
```

ip firewall screen dos-defense syn-flood

This command enables the protection against SYN flood attacks. When the protection is enabled, the amount of TCP packets with the SYN flag set per second for one destination address is limited.

The use of a negative form (no) of the command disables ICMP SYN protection.

Syntax

```
ip firewall screen dos-defense syn-flood { <NUM> } [src-dst]
no ip firewall screen dos-defense syn-flood
```

Parameters

<NUM> – maximum amount of TCP packets with the set SYN flag per second, set in the range of [1..10000].

src-dst – limitation on the amount of TCP packets with the SYN flag set, based on the source and destination addresses.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen dos-defense syn-flood 100 src-dsr
```

ip firewall screen dos-defense udp-threshold

This command enables the protection against UDP flood attacks. When the protection is enabled, the amount of UDP packets per second for one destination address is limited.

The use of a negative form (no) of the command disables UDP flood protection.

Syntax

```
ip firewall screen dos-defense udp-threshold { <NUM> }
```

```
no ip firewall screen dos-defense udp-threshold
```

Parameters

<NUM> – maximum amount of UDP packets per second, set in the range of [1..10000].

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen dos-defense udp-threshold
```

ip firewall screen dos-defense winnuke

This command enables the protection against winnuke attacks. When the protection is enabled, TCP packets with the URG flag set and 139 destination port are blocked.

The use of a negative form (no) of the command disables winnuke attacks protection.

Syntax

```
[no] ip firewall screen dos-defense winnuke
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen dos-defense winnuke
```

ip firewall screen spy-blocking fin-no-ack

This command enables the blocking of TCP packets with the FIN flag set and the ACK flag not set.

The use of a negative form (no) of the command disables the blocking of TCP packets with the FIN flag set and the ACK flag not set.

Syntax

```
[no] ip firewall screen spy-blocking fin-no-ack
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen spy-blocking fin-no-ack
```

ip firewall screen spy-blocking icmp-type destination-unreachable

This command enables the blocking of all 3 type ICMP packets (destination-unreachable) including the packets generated by the router itself.

The use of a negative form of the command (no) disables blocking of ICMP packets of type 3.

Syntax

```
[no] ip firewall screen spy-blocking icmp-type destination-unreachable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen spy-blocking icmp-type destination-unreachable
```

ip firewall screen spy-blocking icmp-type echo-request

This command enables the blocking of all 8 type ICMP packets (echo-request) including the packets generated by the router itself.

The use of a negative form of the command (no) disables blocking of ICMP packets of type 8.

Syntax

```
[no] ip firewall screen spy-blocking icmp-type echo-request
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen spy-blocking icmp-type echo-request
```

ip firewall screen spy-blocking icmp-type reserved

This command enables the blocking of all 2 and 7 type ICMP packets (reserved) including the packets generated by the router itself.

The use of a negative form of the command (no) disables blocking of ICMP packets of type 2 and 7.

Syntax

```
[no] ip firewall screen spy-blocking icmp-type reserved
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen spy-blocking icmp-type reserved
```

ip firewall screen spy-blocking icmp-type source-quench

This command enables the blocking of all 4 type ICMP packets (source quench) including the packets generated by the router itself.

The use of a negative form of the command (no) disables blocking of ICMP packets of type 4.

Syntax

```
[no] ip firewall screen spy-blocking icmp-type source-quench
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen spy-blocking icmp-type source-quench
```

ip firewall screen spy-blocking icmp-type time-exceeded

This command enables the blocking of all 11 type ICMP packets (time exceeded) including the packets generated by the router itself.

The use of a negative form of the command (no) disables blocking of ICMP packets of type 11.

Syntax

```
[no] ip firewall screen spy-blocking icmp-type time-exceeded
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen spy-blocking icmp-type time-exceeded
```

ip firewall screen spy-blocking ip-sweep

This command enables the protection against IP-sweep attacks. When the protection is enabled, if more than 10 ICMP requests from one source arrive within the specified interval, the first 10 requests are dropped by the router and 11th with the following ones are discarded for the remaining interval time.

The use of a negative form (no) of the command disables ip-sweep protection.

Syntax

```
ip firewall screen spy-blocking ip-sweep <THRESHOLD> [ <TIME> ]
no ip firewall screen spy-blocking ip-sweep
```

Parameters

<THRESHOLD> – number of ip sweep attack packets per second, set in the range [1..10000].

<TIME> – blocking time in milliseconds [1..1000000].

Default value

Disabled.

Without specifying a blocking time when turning on, the value is set to 10000.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen spy-blocking ip-sweep 1000
```

ip firewall screen spy-blocking port-scan

This command enables the protection against port scan attacks. If more than 10 TCP packets with the SYN flag arrive to several TCP ports and or more than 10 UDP packets arrive ti several UDP ports of one source within the first specified interval (<threshold>), then this behaviour is recorded as port scan attack and all the following packets of that type are blocked for the second specified time interval (<TIME>).

The use of a negative form (no) of the command disables protection from port scan attacks.

Syntax

```
ip firewall screen spy-blocking port-scan <THRESHOLD> [ <TIME> ]
no ip firewall screen spy-blocking port-scan
```

Parameters

<THRESHOLD> – number of port scan attack packets per second, set in the range [1..10000].

<TIME> – blocking time in milliseconds [1..1000000].

Default value

Disabled.

Without specifying a blocking time when turning on, the value is set to 10000.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen spy-blocking port-scan 100 1000
```

ip firewall screen spy-blocking spoofing

This command enables the protection against ip spoofing attacks. When the protection is enabled, the router checks packets for matching the source address and routing table entries, and in case of mismatch the packet is dropped. For example, if a packet with source address 10.0.0.1/24 arrives to the Gi1/0/1 interface and the given subnet is located after the Gi1/0/2 interface in the routing table, it is considered that the source address has been replaced.

The use of a negative form (no) of the command disables ip spoofing protection.

Syntax

```
[no] ip firewall screen spy-blocking spoofing
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen spy-blocking spoofing
```

ip firewall screen spy-blocking syn-fin

This command enables the blocking of TCP packets, with the SYN and FIN flags set.

The use of a negative form (no) of the command disables the blocking of TCP packets, with the SYN and FIN flags set.

Syntax

```
[no] ip firewall screen spy-blocking syn-fin
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen spy-blocking syn-fin
```

ip firewall screen spy-blocking tcp-all-flags

This command enables the blocking of TCP packets, with all flags or with the set of flags: FIN, PSH, URG. The protection against XMAS attack is provided.

The use of a negative for of the command disables the blocking of TCP packets, with all flags or with the set of flags: FIN,PSH,URG.

Syntax

```
[no] ip firewall screen spy-blocking tcp-all-flag
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen spy-blocking tcp-all-flag
```

ip firewall screen spy-blocking tcp-no-flag

This command enables the blocking of TCP packets with the zero 'flags' field.

The use of a negative form (no) of the command disables the blocking of TCP packets with the zero 'flags' field.

Syntax

```
[no] ip firewall screen spy-blocking tcp-no-flag
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen spy-blocking tcp-no-flag
```

ip firewall screen suspicious-packets icmp-fragment

This command enables the blocking of fragmented ICMP packets.

The use of a negative form (no) of the command disables the blocking of fragmented ICMP packets.

Syntax

```
[no] ip firewall screen suspicious-packets icmp-fragment
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen suspicious-packets icmp-fragment
```

ip firewall screen suspicious-packets ip-fragment

This command enables the blocking of fragmented IP packets.

The use of a negative form (no) of the command disables the blocking of fragmented IP packets.

Syntax

```
[no] ip firewall screen suspicious-packets ip-fragment
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen suspicious-packets ip-fragment
```

ip firewall screen suspicious-packets large-icmp

This command enables the blocking of ICMP packets more than 1024 bytes.

The use of a negative form of the command (no) disables blocking of ICMP packets more than 1024 bytes.

Syntax

```
[no] ip firewall screen suspicious-packets large-icmp
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen suspicious-packets large-icmp
```

ip firewall screen suspicious-packets syn-fragment

This command enables the blocking of fragmented TCP packets with the SYN flag.

The use of a negative form (no) of the command disables the blocking of TCP packets with the SYN flag.

Syntax

`[no] ip firewall screen suspicious-packets syn-fragment`

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen suspicious-packets syn-fragment
```

ip firewall screen suspicious-packets udp-fragment

This command enables the blocking of fragmented UDP packets.

The use of a negative form (no) of the command disables blocking of fragmented UDP packets.

Syntax

`[no] ip firewall screen suspicious-packets udp-fragment`

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen suspicious-packets udp-fragment
```

ip firewall screen suspicious-packets unknown-protocols

This command enables the blocking of packets, with the protocol ID contained in IP header equal to 137 and more.

The use of a negative form (no) of the command disables the blocking of packets, with the protocol ID contained in IP header equal to 137 and more.

Syntax

```
[no] ip firewall screen suspicious-packets unknown-protocols
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall screen suspicious-packets unknown-protocols
```

Management of network attacks notification

ip firewall logging interval

Set the frequency of notification (via SNMP, syslog and in CLI) of detected and blocked network attacks When an attack is detected, a message is logged instantly, but the following alerts about this particular attack will be logged after a specified time interval, if the attack is continuous.

The use of a negative form (no) of the command returns the timer to the default value.

Syntax

```
ip firewall logging interval <NUM>
no ip firewall logging interval
```

Parameters

<NUM> – time interval in seconds [30 .. 2147483647]

Default value

30

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip firewall logging interval 60
```

logging firewall screen detailed

attacks in the CLI.

In addition to the name of the interface from which the packet came, the detailed output shows the source IP address and the destination IP address of the packet, as well as the MAC address of the source that sent the packet.

The use of a negative form (no) of the command disables detailed message output.

Syntax

```
[no] logging firewall screen detailed
```

Parameters

None.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# logging firewall screen detailed
```

logging firewall screen dos-defense

This command enables the mechanism of DoS attacks detection and logging via CLI, syslog and SNMP. In conjunction with the included protection against attacks, an alert will be generated about the reflected DoS attacks.

The use of a negative form (no) of the command disables the mechanism of detection and logging of detected and reflected DoS attacks.

Syntax

```
[no] logging firewall screen dos-defense <ATACK_TYPE>
```

Parameters

<ATACK_TYPE> – DoS attack type, takes the following values:

- icmp-threshold;
- land;
- limit-session-destination;
- limit-session-source;
- syn-flood;
- udp-threshold;
- winnuke.

For a detailed description of DoS attacks, see Managing protection against network attacks.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# logging firewall screen dos-defense land
```

logging firewall screen spy-blocking

This command enables the mechanism of spyware activity detection and logging via CLI, syslog and SNMP. In conjunction with the anti-spyware protection enabled, an alert will be issued about blocked spyware activity.

The use of a negative form (no) of the command disables the mechanism of detection and logging of detected and reflected spyware activity.

Syntax

```
[no] logging firewall screen spy-blocking { <ATACK_TYPE> | icmp-type <ICMP_TYPE> }
```

Parameters

<ATACK_TYPE> – espionage activity type, takes the following values:

- fin-no-ack;
- ip-sweep;
- port-scan;
- spoofing;
- syn-fin;
- tcp-all-flag;
- tcp-no-flag.

<ICMP_TYPE> – icmp type, takes values:

- destination-unreachable;
- echo-request;
- reserved;
- source-quench;
- time-exceeded.

For a detailed description of spy activities, see Managing protection against network attacks.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# logging firewall screen spy-blocking icmp-type echo-request
```

logging firewall screen suspicious-packets

This command enables the mechanism of non-standard packets detection and logging via CLI, syslog and SNMP. In conjunction with the included protection against non-standard packets, a notification will also be issued about blocking non-standard packets.

The use of a negative form (no) of the command disables the mechanism of detection and logging of detected and blocked non-standard packets.

Syntax

```
[no] logging firewall screen suspicious-packets <PACKET_TYPE>
```

Parameters

<PACKET_TYPE> – specialized packets type, takes the following values:

- icmp-fragment;
- ip-fragment;

- large-icmp;
- syn-fragment;
- udp-fragment;
- unknown-protocols.

For a detailed description of protection against non-standard packets, see Managing protection against network attacks.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# logging firewall screen suspicious-packets icmp-fragment
```

show ip firewall screens counters

This command allows you to view statistics on detected network attacks.

Syntax

`show ip firewall screens counters`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

ROOT

Example

```
esr# show ip firewall screens counters
DDoS:
    Destination limit screen:  --
    Source limit screen:       --
    ICMP threshold screen:     1
    UDP threshold screen:      --
    SYN flood screen:          0
    Land attack screen:        1
    Winnuke attack screen:     --
Suspicious packets:
    ICMP fragmented screen:    --
    UDP fragmented screen:     --
    Large ICMP screen:         4
    Fragmented SYN screen:     --
    Unknown protocol screen:   --
    Fragmented IP screen:      --
Spying:
    Port scanning screen:      --
    IP sweep secreen:          --
    SYN-FIN screen:            --
    TCP all flags screen:      --
    FIN no ACK screen:         --
    TCP no flags screen:       --
    Spoofing screen:           --
ICMP types:
    ICMP reserved screen:      --
    ICMP quench screen:        --
    ICMP echo request screen:  --
    ICMP time exceeded screen: --
    ICMP unreachable screen:   --
```

Filtering management

- default action
- description
- filter
- log enable
- ip http profile
- ip http proxy
- ip http proxy aaa das-profile
- ip http proxy listen-ports
- ip http proxy redirect-port
- ip http proxy server-url
- ip https proxy
- ip https proxy listen-ports
- ip https proxy redirect-port
- urls local
- urls remote

default action

The command sets an action for unauthorized users' traffic processing .

The use of a negative (no) form of the command deletes an action for unauthorized users' traffic processing.

Syntax

```
default action <ACTION> { redirect-url <REDIRECT-URL>}
```

```
no default action
```

Parameters

<ACTION> – an action for traffic that is not described in a profile. Possible values:

- permit – permit HTTP requests;
- deny – block HTTP requests;
- redirect – redirect requests to a certain URL address.

<REDIRECT-URL> – URL address for HTTP requests redirection. Specified in 'redirect' mode only.

Default value

Not specified.

Required privilege level

15

Command mode

CONFIG-PROXY-PROFILE

Example

```
esr(config-profile)# default action deny
```

description

This command specifies the description of a filtering profile.

The use of a negative form (no) of the command removes description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – interface description, set by a string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-PROXY-PROFILE

Example

```
esr(config-profile)# description "deny any"
```

filter

The command enables traffic filtering mode.

The use of a negative form (no) of the command disables traffic filtering mode.

Syntax

```
[no] filter <DATA-TYPE>
```

Parameters

<DATA-TYPE> – a type of data filtered. May take the following values:

- activex
- cookie
- js

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-PROXY-PROFILE

Example

```
esr(config-profile)# filter cookie
```

log enable

This command enables proxy profile logging.

The use of a negative form (no) of the command enables proxy profile logging.

Syntax

`[no] log enable`

Parameters

None.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-PROXY-PROFILE

Example

```
esr(config-proxy-profile)# log enable
```

ip http profile

The command creates an HTTP traffic filtering profile and enables profile configuration mode.

The use of a negative form of the command deletes an HTTP traffic filtering profile.

Syntax

```
[no] ip http profile <NAME>
```

Parameters

<NAME> – a name of an HTTP traffic filtering profile set by a string of [1..31] characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip http profile HTTP-BLOCKED
```

ip http proxy

The command assigns an HTTP traffic filtering profile to an interface or a tunnel.

The use of a negative form of the command deletes an HTTP traffic filtering profile.

Syntax

```
ip http proxy <NAME>
no ip http proxy
```

Parameters

<NAME> – profile name. Set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-CELLULAR-MODEM

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-LT

CONFIG-PPPOE

Example

```
esr(config-cellular-modem)# ip http proxy HTTP-BLOCKED
```

ip http proxy aaa das-profile

The command sets a profile for dynamic authorization servers (DAS) to which CoA requests on URL lists modification will be sent by PCRF.

The use of a negative (no) form of the command removes the DAS profile.

Syntax

```
ip http proxy aaa das-profile <NAME>
no ip http proxy aaa das-profile
```

Parameters

<NAME> – DAS profile name, set by the string of up to 31 characters.

Default value

Profile is not specified.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip http proxy aaa das-profile my_server
```

ip http proxy listen-ports

The command creates a list of TCP ports (in addition to 80 and 8080) for HTTP requests filtering.

The use of a negative (no) form of the command removes a list created before.

Syntax

```
[no] ip http proxy listen-ports <SERVICE_OBJ_GROUP_NAME>
```

Parameters

<SERVICE_OBJ_GROUP_NAME> – port profile (object-group service). Set by the string of up to 31 characters.

Default value

80, 8080

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip http proxy listen-ports HTTP-ADD
```

ip http proxy redirect-port

This command defines the base HTTP Proxy server port on the router. The number of ports used corresponds to the number of processor cores.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip http proxy redirect-port <PORT>
no ip http proxy redirect-port
```

Parameters

<PORT> – port number, set in the range of [1..65535].

Default value

3128

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip proxy http redirect-port 3001
```

ip http proxy server-url

The command specifies a URL address of a server from which a filtering list will be received.

The use of a negative form (no) of the command removes a specified server URL.

Syntax

```
[no] ip http proxy server-url <URL>
```

Parameters

<URL> – URL address of a resource from which a file containing a filtering list should be downloaded. Set by a string of [8..255] characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip http proxy server-url http://domain.example
```

ip https proxy

The command assigns an HTTP request filtering profile to an interface or a tunnel.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip https proxy <NAME>
no ip https proxy
```

Parameters

<NAME> – profile name. Set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-CELLULAR-MODEM

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-LT

CONFIG-PPPOE

Example

```
esr(config-cellular-modem)# ip https proxy HTTPS-BLOCKED
```

ip https proxy listen-ports

The command creates a list of TCP ports (in addition to 433) for HTTP requests filtering.

The use of a negative (no) form of the command removes a list created before.

Syntax

```
[no] ip https proxy listen-ports <SERVICE_OBJ_GROUP_NAME>
```

Parameters

<SERVICE_OBJ_GROUP_NAME> – port profile (object-group service). Set by the string of up to 31 characters.

Default value

433

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip https proxy listen-ports HTTPS-ADD
```

ip https proxy redirect-port

This command defines the base HTTPS Proxy server port on the router. The number of ports used corresponds to the number of processor cores.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip https proxy redirect-port <PORT>
```

```
no ip https proxy redirect-port
```

Parameters

<PORT> – port number, set in the range of [1..65535].

Default value

3129

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip proxy https redirect-port 3001
```

urls local

The command adds a local list of URL addresses and actions for this list to a filtering profile.

The use of a negative form (no) of the command sets the default value.

Syntax

```
urls local <URL_OBJ_GROUP_NAME> action { deny | permit | redirect redirect-url <REDIRECT-URL> }
```

```
no urls local
```

Parameters

<URL_OBJ_GROUP_NAME> – a list of URLs an action will be applied to;

permit – pass http requests to addresses described in the specified URL list;

deny – block http requests to addresses described in the specified URL list;

redirect – redirect requests to addresses described in the specified URL list to a specific url address.

<REDIRECT-URL> – URL address for HTTP requests redirection. Specified in 'redirect' mode only.

Required privilege level

15

Command mode

CONFIG-PROFILE

Example

```
esr(config-profile)# urls local OGU_DENY action deny
```

urls remote

The command adds a link to a remote list containing URL addresses and actions applied to this list.

The use of a negative form (no) of the command sets the default value.

Syntax

```
urls remote <FILE-NAME> action { deny | permit | redirect redirect-url <REDIRECT-URL> }
no urls remote
```

Parameters

<FILE-NAME> – a name of a file that contains a URL list and is located on a server specified in ip http proxy server-url (see ip http proxy server-url). File name is set by a string of up to 31 characters.

permit – pass http requests to addresses described in the file with specified name;

deny – block http requests to addresses described in the file with specified name;

redirect – redirect requests to addresses described in the file with specified name to a specific url address.

<REDIRECT-URL> – URL address for HTTP requests redirection. Specified in 'redirect' mode only.

Required privilege level

15

Command mode

CONFIG-PROFILE

Example

```
esr(config-profile)# urls remote http-deny action deny
```

Intrusion Prevention System (IPS/IDS) configuration

- IPS/IDS general commands
    - description
    - enable
    - show security ips content-provider
    - show security ips content-provider rules-info
    - show security ips counters
    - show security ips user-server
    - update security ips content-provider rules
    - update security ips content-provider rules-info
    - update security ips user-server rules
- IPS/IDS policy configuration
    - category
    - external network-group
    - protect network-group
    - rules action
    - rules count
    - security ips policy
    - vendor
- IPS configuration
    - logging ips severity
    - logging storage-path
    - security ips
    - performance max
    - policy
    - service-ips enable
- Configuring auto-updating of IPS/IDS rules distributed under a commercial license
    - content-provider
    - host address
    - host port
    - reboot
    - storage-device
    - upgrade interval
- Configuration of IPS/IDS rules autoupdate from external sources
    - auto-upgrade
    - upgrade interval
    - url
    - user-server
- User IPS/IDS rules configuration
    - action
    - destination-address
    - destination-port
    - direction
    - ip dscp
    - ip ftp command
    - ip ftp-data command
    - ip http
    - ip http content-filter
    - ip http method
    - ip icmp code
    - ip icmp code comparison-operator
    - ip icmp id

    - ip icmp sequence-id
    - ip icmp type
    - ip icmp type comparison-operator
    - ip protocol-id
    - ip tcp acknowledgment-number
    - ip tcp sequence-id
    - ip tcp window-size
    - ip ttl
    - ip ttl comparison-operator
    - meta classification-type
    - meta log-message
    - payload content
    - payload data-size
    - payload data-size comparison-operator
    - payload depth
    - payload no-case
    - payload offset
    - protocol
    - rule
    - security ips-category user-defined
    - source-address
    - source-port
    - threshold count
    - threshold second
    - threshold track
    - threshold type
- Extended user rules configuration
    - rule-advanced
    - rule-text

IPS/IDS general commands

description

This command changes the description.

The use of a negative form (no) of the command removes description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – description, set by the string of up to 255 characters.

Required privilege level

10

Command mode

CONFIG-IPS-CATEGORY

CONFIG-IPS-CATEGORY-RULE

CONFIG-IPS-CATEGORY-RULE-ADVANCED

CONFIG-IPS-POLICY

CONFIG-IPS-UPGRADE-USER-SERVER

CONFIG-CONTENT-PROVIDER

Example

```
esr(config-ips-upgrade-user-server)# description "Etnetera aggressive IP blacklist"
```

enable

This command activates the IPS/IDS service and its rules.

The use of a negative form (no) of the command deactivates the IPS/IDS service.

Syntax

`[no] enable`

Parameters

The command does not contain parameters.

Default value

IPS/IDS service is not activated.

Required privilege level

15

Command mode

CONFIG-IPS

CONFIG-IPS-CATEGORY-RULE

CONFIG-IPS-CATEGORY-RULE-ADVANCED

CONFIG-CONTENT-PROVIDER

Example

```
esr(config-ips)# enable
```

show security ips content-provider

This command allows to view information about updates of IPS/IDS rules distributed under a commercial license.

Syntax

```
show security ips content-provider
```

Required privilege level

10

Command mode

ROOT

Example

```
esr# show security ips content-provider
 Server: content-provider
        Last MD5 of received files:        93633ab9a73248ea50d58c25b1ac806c
        Next update: 06 October 2020 12:27:40
```

show security ips content-provider rules-info

This command allows to view information about categories of IPS/IDS rules available under the current commercial license. If there is no valid license, the list will be empty.

Syntax

```
show security ips content-provider rules-info
```

Required privilege level

10

Command mode

ROOT

Example

```
esr# show security ips content-provider rules-info
Vendor : kaspersky
    Category : IoTURLsDF
        Count of rules : 8000
        Description : Kasperksy Lab IoTURLsDF feed
              IoTURLsDF URL feed - a set of URLs with context covering malware that infects IoT
(Internet of Things) devices
    Category : MaliciousHashDF
        Count of rules : 1
        Description : Kasperksy Lab MaliciousHashDF feed
              Malicious Hash feed - a set of hashes of malicious objects
    Category : PhishingURLsDF
        Count of rules : 11167
        Description : Kasperksy Lab PhishingURLsDF feed
              Phishing URL feed - a set of URLs with context that cover phishing websites and
web pages
```

show security ips counters

This command scans IPS/IDS service counters.

Syntax

```
show security ips counters
```

Required privilege level

10

Command mode

ROOT

Example

```
esr# show security ips counters
TCP flows processed : 34687
Alerts generated : 456
Blocked by ips engine : 78
Accepted by ips engine : 1356436
```

show security ips user-server

This command allows to view information about IPS/IDS rule updates from user update servers.

Syntax

```
show security ips user-server [<WORD>]
```

Parameters

<WORD> – server name, specified by the string from 1 to 64 characters long.

Required privilege level

10

Command mode

ROOT

Example

```
esr# sh security ips user-server
Server name                        Files MD5                          Next update
--------------------------------   --------------------------------
--------------------------------
content-provider                   93633ab9a73248ea50d58c25b1ac806c   06 October 2020 12:27:40
TH                                 919f51bdf44052bfc0953362aef11c0d   06 October 2020 12:36:40
Traffic-ID                         e5e2f6472a397227c0d96f5df430a207   06 October 2020 12:36:40
Aggressive                         cfc3547b50f3f9fec366ba5a1e51cd1f   06 October 2020 12:36:40
JA3-Fingerprint                    439aa6e57c66826b92337672937d505b   05 October 2020 16:51:40
C2-Botnet                          39e118bd3884b3dc1df4ca3a03c05df1   05 October 2020 16:51:40
SSL-BlackList                      1d9c969f25791b9ee8c8c0ab8449d849   05 October 2020 16:51:40
ET-Open                            d53d92248a1f7cdc040d669a76cf27bc   06 October 2020 12:36:40
```

update security ips content-provider rules

This command initiates a forced update of IPS/IDS rules distributed under a commercial license.

The actual start of the rule updating procedure occurs with some delay after the command is entered. The maximum delay is 5 minutes.

Syntax

```
update security ips content-provider rules
```

Required privilege level

15

Command mode: update security ips content-provider rules

ROOT

Example

```
esr# update security ips content-provider rules
```

update security ips content-provider rules-info

This command initiates a forced request for information about categories of IPS/IDS rules available under the current commercial license.

The actual start of the rule updating procedure occurs with some delay after the command is entered. The maximum delay is 5 minutes.

Syntax

```
update security ips content-provider rules-info
```

Required privilege level

15

Command mode

ROOT

Example

```
esr# update security ips content-provider rules-info
```

update security ips user-server rules

This command initiates a forced update of IPS/IDS rules from the user update server.

The actual start of the rule updating procedure occurs with some delay after the command is entered. The maximum delay is 5 minutes.

Syntax

```
update security ips user-server rules <WORD>
```

Parameters

<WORD> – server name, specified by the string from 1 to 64 characters long.

Required privilege level

15

Command mode

ROOT

Example

```
esr# update security ips user-server rules ET-Open
```

IPS/IDS policy configuration

category

This command specifies the category of IPS/IDS rules of a particular vendor, distributed under a commercial license, and enters the configuration mode for that category

The use of a negative form (no) of the command removes the configured category from the IPS/IDS service settings.

Syntax

```
category <CATEGORY>
```

```
no category { <CATEGORY> | all }
```

Parameters

<CATEGORY> – rule category.

You can see the list of available categories in the context tooltip or with a command:

show security ips content-provider rules-info

Required privilege level

15

Command mode

CONFIG-IPS-VENDOR

Example

```
esr(config-ips-vendor)# category MobileBotnetCAndCDF
```

external network-group

This command sets the IP address profile, which the IPS/IDS service will consider unreliable.

The IP address profile must be pre-created.

The use of a negative form (no) of the command removes the configured profile from the IPS/IDS service settings.

Syntax

```
external network-group <OBJ-GROUP-NETWORK-NAME>
```

```
no external network-group
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – IP addresses profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-IPS-POLICY

Example

```
esr(config-ips-policy)# external network-group WAN
```

protect network-group

This command sets the IP address profile that the IPS/IDS service will protect.

The IP address profile must be pre-created.

The use of a negative form (no) of the command removes the configured profile from the IPS/IDS service settings.

Syntax

```
protect network-group <OBJ-GROUP-NETWORK-NAME>
no protect network-group
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – IP addresses profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-IPS-POLICY

Example

```
esr(config-ips-policy)# protect network-group LAN
```

rules action

The command specifies the action that should be applied for the traffic meeting rules of this category.

The use of a negative form (no) of the command removes an assigned action.

The command applies only to rules distributed under a commercial license.

Syntax

```
rules action { alert | reject | pass | drop }
```

```
no rules action
```

Parameters:

- alert – traffic is allowed and the IPS/IDS service generates a message;
- reject – traffic is prohibited. If it is TCP traffic, a TCP-RESET packet is sent to the sender and recepient, for the rest of the traffic type, an ICMP-ERROR packet is sent. IPS/IDS service generates a message;
- pass – traffic transfer is permitted;
- drop – traffic is prohibited and the IPS/IDS service generates a message.

Required privilege level

15

Command mode

CONFIG-IPS-VENDOR-CATEGORY

Example

```
esr(config-ips-vendor-category)# rules action drop
```

rules count

This command specifies the effective number of rules of a given category that the IPS/IDS system will operate with

The use of a negative form (no) of the command removes an assigned action.

The command applies only to rules distributed under a commercial license.

Syntax

```
rules count <COUNT>
```

```
no rules count
```

Parameters:

<COUNT> – number of rules. The minimum value is 1, the maximum value depends on the category of rules.

The maximum number of rules by category can be seen in the context hint or with the command:

show security ips content-provider rules-info

Required privilege level

15

Command mode

CONFIG-IPS-VENDOR-CATEGORY

Example

```
esr(config-ips-vendor-category)# rules count 8000
```

security ips policy

This command creates an IPS/IDS service settings policy with a specific name and switches to the policy configuration mode.

The use of a negative form (no) of the command removes the configured policy of the IPS/IDS service settings.

Syntax

```
[no] security ips policy <POLICY_NAME>
```

Parameters

<POLICY_NAME> – IPS/IDS service policy name, specified by a string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# security ips policy OFFICE
```

vendor

This command identifies the vendor of IPS/IDS rules distributed under a commercial license and enters the configuration mode for that vendor.

The use of a negative form (no) of the command removes the configured vendor from the IPS/IDS service settings.

Syntax

```
vendor <VENDOR>
```

```
no vendor <CATEGORY>
```

Parameters

<VENDOR> – rule vendor.

You can see the list of available vendors in the context tooltip or with a command:

show security ips content-provider rules-info

Required privilege level

15

Command mode

CONFIG-IPS-POLICY

Example

```
esr(config-ips-policy)# vendor kaspersky
```

IPS configuration

logging ips severity

This command sets the message severity level for logging IPS/IDS events.
The use of a negative form (no) of the command sets the default value.

Syntax

```
logging ips severity <SEVERITY>
```

```
no logging ips severity
```

Parameters

<SEVERITY> – message importance level, takes values (in order of decreasing importance):

- emerg – critical error has occurred in the system, the system is not operational;
- alert – alarms, immediate intervention by staff;
- crit – critical system status, event reporting;
- error – error messages;
- warning – warnings, non-emergency messages;
- notice – messages about important system events;
- info – system information messages;
- debug – debugging messages provide the user with information to correctly configure the system;
- none – disables the output of syslog messages to the console.

Default value

info

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# logging ips severity error
```

logging storage-path

This command sets the name and path of the directory on the external drive to which the log files of the IPS/IDS service in the EVE format (elasticsearch) will be written.

The use of a negative form (no) of the command stops recording log files.

Syntax

```
logging storage-path <PATH>
no logging storage-path
```

Parameters

<PATH> – the name and path of the directory on the external drive in format of:

usb://usb_name:/[FILE]/
mmc://mmc_name:/[FILE]/

Required privilege level

15

Command mode

CONFIG-IPS

Example

```
esr(config-ips)# logging storage-path usb://DATA/Log/
```

security ips

This command creates an IPS/IDS service profile and switch to its configuration mode.

Syntax

```
security ips
```

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# security ips
```

performance max

This command allows the IPS/IDS service to use all of the device's resources for maximum performance. It is recommended to use when the device is used exclusively as IPS/IDS. It is not recommended to use when, in addition to IPS/IDS, the device performs other functions (routing, BRAS, etc.).

The use of a negative form (no) of the command frees up part of the device's resources for use by other services.

Syntax

`[no] performance max`

Required privilege level

15

Command mode

CONFIG-IPS

Example

```
esr(config-ips)# perfomance max
```

policy

This command assigns the previously created IPS/IDS service settings policy.

The use of a negative form (no) of the command removes the assigned policy of the IPS/IDS service settings.

Syntax

`policy <POLICY_NAME>`

`no policy`

Parameters

<POLICY_NAME> – IPS service policy name, specified by a string of up to 32 characters.

Required privilege level

15

Command mode

CONFIG-IPS

Example

```
esr(config-ips)# policy OFFICE
```

service-ips enable

This command is used to enable the IPS/IDS service on the network interface.

The use of a negative form (no) of the command disables the IPS/IDS service on the network interface.

Syntax

```
[no] service-ips enable
```

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# service-ips enable
```

Configuring auto-updating of IPS/IDS rules distributed under a commercial license

content-provider

This command switches to the configuration mode of the source of rule updates distributed under a commercial license.

Syntax

```
content-provider
```

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# content-provider
```

host address

This command specifies the address of the server for rule updates distributed under a commercial license.

Syntax

`host address { <ADDR> | <IPV6-ADDR> | <HOSTNAME> }`

Parameters

<ADDR> – device IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – device IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<HOSTNAME> – user DNS name, set by the string of up to 255 characters;

Required privilege level

15

Command mode

CONFIG-CONTENT-PROVIDER

Example

```
esr(config-content-provider)# host address edm.eltex-co.ru
```

host port

This command specifies the TCP port number of the server for rule updates distributed under a commercial license.

Syntax

`host port <PORT>`

Parameters

<PORT> – TCP port number, may take values [1..65535];

Required privilege level

15

Command mode

CONFIG-CONTENT-PROVIDER

Example

```
esr(config-content-provider)# host port 8098
```

reboot

This command sets the time to reboot the device when the system license is received. The device reboots the first time it connects to a commercially licensed rule update server.

If you have an already operating IPS/IDS license, there is no reboot.

Syntax

```
reboot { immediately | time <TIME> }
```

Parameters

immediately – reboot immediately after receiving a license;

time <TIME> – restart at a specified time <TIME>;

<TIME> – reboot time in format of HH:MM:SS.

Required privilege level

15

Command mode

CONFIG-CONTENT-PROVIDER

Example

```
esr(config-content-provider)# reboot time 05:00:00
```

storage-device

This command specifies the name of the external drive on which encrypted IPS/IDS rules distributed under a commercial license will be stored.

The use of a negative form (no) of the command stops rule saving.

Syntax

```
storage-device <PATH>
no storage-device
```

Parameters

<PATH> – name of external drive in format of:

usb://usb_name:/
mmc://mmc_name:/

Required privilege level

15

Command mode

CONFIG-CONTENT-PROVIDER

Example

```
esr(config-content-provider)# storage-device usb://DATA
```

upgrade interval

The command specifies the frequency with which the device will check for updates of IPS/IDS rules distributed under a commercial license.

The use of a negative form (no) of the command sets the default value.

Syntax

```
upgrade interval <HOURS>
no upgrade interval
```

Parameters

<HOURS> – update interval in hours, from 1 to 240.

Default value

24

Required privilege level

15

Command mode

CONFIG-CONTENT-PROVIDER

Example

```
esr(config-content-provider)# upgrade interval 36
```

Configuration of IPS/IDS rules autoupdate from external sources

auto-upgrade

This command switches to the configuration mode of the sources of rule updates for the service.

Syntax

```
auto-upgrade
```

Required privilege level

15

Command mode

CONFIG-IPS

Example

```
esr(config-ips)# auto-upgrade
```

upgrade interval

This command sets the frequency with which the device will check for the updates for IPS/IDS rules and/or IPS/IDS classifier file for this url.

The use of a negative form (no) of the command sets the default value.

Syntax

```
upgrade interval <HOURS>
```

```
no upgrade interval
```

Parameters

<HOURS> – update interval in hours, from 1 to 240.

Default value

24

Required privilege level

15

Command mode

CONFIG-IPS-UPGRADE-USER-SERVER

Example

```
esr(config-ips-upgrade-user-server)# upgrade interval 36
```

url

The command specifies URL link.

The use of a negative form (no) of the command removes the link from the IPS/IDS rule update source configuration.

Syntax

```
url <URL>
```

```
no url
```

Parameters

<URL> – text field containing URL link of 8-255 characters length.

As an URL-links can be specified:

- rule file with the .rule extension.
- rule classifier file named classification.config
- directory on the server containing rule files and/or rule classifier file.

Required privilege level

15

Command mode

CONFIG-IPS-UPGRADE-USER-SERVER

Example

```
esr(config-ips-upgrade-user-server)# url https://rules.emergingthreats.net/open/suricata-4.0/
rules/
```

user-server

This command sets the name of the user IPS/IDS rule update server and switches to the configuration mode of the user update server settings.

The use of a negative form (no) of the command removes the user IPS/IDS rule update server and all the rules received from this server.

Syntax

```
user-server <WORD>
```

```
no user-server { <WORD> | all }
```

Parameters

<WORD> – server name, specified by the string from 1 to 64 characters long.

Required privilege level

15

Command mode

CONFIG-IPS-AUTO-UPGRADE

Example

```
esr(config-ips-auto-upgrade)# user-server ET-Open
```

User IPS/IDS rules configuration

action

The command specifies the action that should be applied for the traffic meeting this requirements.

The use of a negative form (no) of the command removes an assigned action.

Syntax

```
action { alert | reject | pass | drop }
```

```
no action
```

Parameters:

- alert – traffic is allowed and the IPS/IDS service generates a message;
- reject – traffic is prohibited. If it is TCP traffic, a TCP-RESET packet is sent to the sender and recepient, for the rest of the traffic type, an ICMP-ERROR packet is sent. IPS/IDS service generates a message;
- pass – traffic transfer is permitted;
- drop – traffic is prohibited and the IPS/IDS service generates a message.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# action reject
```

destination-address

The command sets destination IP addresses for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
destination-address { ip <ADDR> | ip-prefix <ADDR/LEN> | object-group <OBJ_GR_NAME> |
policy-object-group { protect | external } | any }
```

```
no destination-address
```

Parameters

<ADDR> – receiver IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<ADDR/LEN> – IP subnet, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32];

<OBJ_GR_NAME> – name of IP addresses profile that contains destination IP address, set by the string of up to 31 characters;

destination-address policy-object-group protect – sets protect addresses defined in IPS/IDS policy as destination addresses;

destination-address policy-object-group external – sets external addresses defined in IPS/IDS policy as destination addresses;

When specifying the 'any' value, the rule will be triggered for any source IP address.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# destination-address ip 10.10.10.1
```

destination-port

The command sets the number of source TCP/UDP port for which the rule should work.

The use of a negative form (no) of the command removes the assignment.

Syntax

```
destination-port { any | <PORT> | object-group <OBJ-GR-NAME> }
no destination-port
```

Parameters

<PORT> – number of destination TCP/UDP port, takes values of [1..65535];

<OBJ_GR_NAME> – recepient TCP/UDP ports profile name, set by the string of up to 31 characters.

When specifying the 'any' value, the rule will be triggered for any source TCP/UDP port.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# destination-port 22
```

direction

This command sets traffic direction for which the rule should be triggered.

The use of a negative form (no) of the command removes the assignment.

Syntax

```
direction { one-way | round-trip }
no direction
```

Parameters

- one-way – traffic is transmitted in one direction;
- round-trip – traffic is transmitted in both directions.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# direction one-way
```

ip dscp

This command sets the value of the DSCP code, the traffic of which will be processed in this rule.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip dscp <DSCP>
[no] ip dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip dscp 8
```

ip ftp command

This command sets the FTP keyword values for which the rule should be triggered.

This command is applicable only for protocol ftp value.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip ftp command <COMMAND>
[no] ip ftp command
```

Parameters

<COMMAND> – can take the following values:

- <retr> – download file;
- <stor> – upload file;
- <mkd> – create directory;
- <rmd> – remove directory;
- <appe> – add to the end of the file (with creation);
- <allo> – allocate space on disk;
- <dele> – delete file.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# protocol ftp
esr(config-ips-category-rule)# ip ftp command allo
```

ip ftp-data command

This command sets the FTP-DATA keyword values for which the rule should be triggered.

This command is applicable only for protocol ftp-data value.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip ftp-data command <COMMAND>
```

```
[no] ip ftp-data command
```

Parameters

<COMMAND> – can take the following values:

- <retr> – download file;
- <stor> – upload file;
- <appe> – add to the end of the file (with creation).

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# protocol ftp-data
esr(config-ips-category-rule)# ip ftp-data command stor
```

ip http

This command sets the HTTP keyword values for which the rule should be triggered.

This command is applicable only for protocol http value.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip http <COMMAND>
```

```
[no] ip http
```

Parameters

<COMMAND> – can take the following values:

- accept;
- accept-enc;
- accept-lang;
- client-body;
- connection;
- content-len;
- content-type;
- cookie;
- file-data;
- header;
- header-names;
- host;

- protocol;
- referer;
- request-line;
- response-line;
- server-body;
- start;
- stat-code;
- stat-msg;
- uri;
- urilen <VALUE>;
- urilen comparison-operator { greater-than | less-than};
- user-agent.

The values and application of the HTTP keywords are detailed described in the SNORT 2.X/Suricata 4.X documentation.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# payload content «HTTP/1.0»
esr(config-ips-category-rule)# ip http protocol
```

ip http content-filter

This command is used to assign a content filtering category profile. The current rule will be triggered for http sites that belong to the categories set in this profile.

The content filtering profile must be pre-created.

This command is applicable only for protocol http value.

The use of a negative form (no) of the command cancels the assignment.

Syntax

`ip http content-filter <NAME>`

`[no] ip http content-filter`

Parameters

<NAME> – name of the content filtering profile, specified as a string of up to 31 characters.

any – rule will trigger for http sites of any category.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip http content-filter Black-List
```

ip http method

This command sets the values of the http access method for which the rule should be triggered.

This command is applicable only for protocol http value.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip http method <COMMAND>
[no] ip http method
```

Parameters

<COMMAND> – can take the following values:

- <GET> – requests a resource submission. Requests using this method can only retrieve data;
- <HEAD> – requests the resource in the same way as the GET method, but without the response body;
- <POST> – is used to send subjects to a specific resource;
- <PUT> – replaces all current resource views with request data;
- <DELETE> – deletes the specified resource;
- <CONNECT> – establishes a «tunnel» to the server defined by the resource;
- <OPTIONS> – used to describe the parameters of the connection to the resource;
- <TRACE> – performs a call of the returned test message from the resource;
- <PATCH> – used to partially modify the resource.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip http method get
```

ip icmp code

This command sets the ICMP CODE value at which the rule will be triggered.

This command is applicable only for protocol icmp value.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip icmp code <CODE>
[no] ip icmp code
```

Parameters

<CODE> – ICMP CODE value, takes a value in the range [0..255].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip icmp code 5
```

ip icmp code comparison-operator

Comparison operator for ip icmp code command. Applicable only in conjunction with this command.

The use of a negative form (no) of the command cancels the comparison.

Syntax

```
ip icmp code comparison-operator { greater-than | less-than }
[no] ip icmp code comparison-operator
```

Parameters

- greater-than – greater than;
- less-than – less than.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip icmp code 5
esr(config-ips-category-rule)# ip icmp code comparison-operator less-than
```

ip icmp id

This command sets the ICMP ID value at which the rule will be triggered.

This command is applicable only for protocol icmp value.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip icmp id <ID>
[no] ip icmp id
```

Parameters

<ID> – ICMP ID value, takes a value in the range [0..65535].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip icmp id 65000
```

ip icmp sequence-id

This command sets the ICMP sequence-ID value at which the rule will be triggered.

This command is applicable only for protocol icmp value.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip icmp sequence-id <SEQ-ID>
[no] ip icmp sequence-id
```

Parameters

<SEQ-ID> – ICMP Sequence-ID value, takes a value in the range [0..4294967295].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip icmp sequence-id 8388608
```

ip icmp type

This command sets the ICMP TYPE value at which the rule will be triggered.

This command is applicable only for protocol icmp value.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip icmp type <TYPE>
[no] ip icmp type
```

Parameters

<TYPE> – ICMP TYPE value, takes a value in the range [0..255].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip icmp type 12
```

ip icmp type comparison-operator

Comparison operator for ip icmp type command. Applicable only in conjunction with this command.

The use of a negative form (no) of the command cancels the comparison.

Syntax

```
ip icmp type comparison-operator { greater-than | less-than }
[no] ip icmp type comparison-operator
```

Parameters

- greater-than – greater than;
- less-than – less than.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip icmp type 14
esr(config-ips-category-rule)# ip icmp code comparison-operator greater-than
```

ip protocol-id

This command sets the IP identification number, the traffic of which will be processed in this rule.

This command is applicable only for protocol any value.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip protocol-id <ID>
[no] ip protocol-id
```

Parameters

<ID> – IP identification number [1..255].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip protocol-id 250
```

ip tcp acknowledgment-number

This command sets the TCP Acknowledgment-Number at which the rule will be triggered.

This command is applicable only for protocol tcp value

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip tcp acknowledgment-number <ACK-NUM>
[no] ip tcp acknowledgment-number
```

Parameters

<ACK-NUM> – TCP Acknowledgement-Number value, takes a value in the range [0..4294967295].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip tcp acknowledgment-number 32
```

ip tcp sequence-id

This command sets the TCP Sequence-ID value at which the rule will be triggered.

This command is applicable only for protocol tcp value

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip tcp sequence-id <SEQ-ID>
[no] ip tcp sequence-id
```

Parameters

<SEQ-ID> – TCP Sequence-ID value, takes a value in the range [0..4294967295].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip tcp sequence-id 2542
```

ip tcp window-size

This command sets the TCP Window Size at which the rule will be triggered.

This command is applicable only for protocol tcp value

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip tcp window-size <SIZE>
[no] ip tcp window-size
```

Parameters

<SIZE> – TCP Window-Size value, takes a value in the range [1..65535]

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip tcp window-size 50
```

ip ttl

This command sets the value of the IP packet lifetime, the traffic of which will be processed in this rule.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
ip ttl <TTL>
[no] ip ttl
```

Parameters

<TTL> – IP packet life time, takes value in the range of [1..255].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip ttl 8
```

ip ttl comparison-operator

Comparison operator for ip ttl command. Applicable only in conjunction with this command.

The use of a negative form (no) of the command cancels the comparison.

Syntax

```
ip ttl comparison-operator { greater-than | less-than }
```

```
[no] ip ttl comparison-operator
```

Parameters

- greater-than – greater than;
- less-than – less than.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# ip ttl 5
esr(config-ips-category-rule)# ip ttl comparison-operator less-than
```

meta classification-type

This command defines the classification of the event that the IPS/IDS service will generate when the rule will be triggered.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
meta classification-type { not-suspicious | unknown | bad-unknown | attempted-recon | successful-recon-limited | successful-recon-largescale | attempted-dos | successful-dos | attempted-user | unsuccessful-user | successful-user | attempted-admin | successful-admin | rpc-portmap-decode | shellcode-detect | string-detect | suspicious-filename-detect | suspicious-login | system-call-detect | tcp-connection | trojan-activity | unusual-client-port-connection | network-scan | denial-of-service | non-standard-protocol | protocol-command-decode | web-application-activity | web-application-attack | misc-activity | misc-attack | icmp-event | inappropriate-content | policy-violation | default-login-attempt }
```

```
[no] meta classification-type
```

Parameters

- not-suspicious – not suspicious traffic;
- unknown – unknown traffic;
- bad-unknown – potentially bad traffic;
- attempted-recon – information leak attempt;
- successful-recon-limited – information leak;
- successful-recon-largescale – large-scale information leak;
- attempted-dos – denial of service attempt;

- successful-dos – denial of service;
- attempted-user – attempt to obtain user privileges;
- unsuccessful-user – unsuccessful attempt to obtain user privileges;
- successful-user – successful attempt to obtain user privileges;
- successful-admin – successful attempt to obtain admin privileges;
- successful-admin – successful attempt to obtain admin privileges;
- rpc-portmap-decode – RPC request decoding;
- shellcode-detect – executable code detected;
- string-detect – suspicious string detected;
- suspicious-filename-detect – suspicious filename was detected;
- suspicious-login – attempt to log in using a suspicious username was deteceted;
- system-call-detect – system call was detected;
- tcp-connection – TCP connection was detected;
- trojan-activity – network Trojan was detected;
- unusual-client-port-connection – the client used an unusual port;
- network-scan – network scan was detected;
- denial-of-service – denial of service attack was detected;
- non-standard-protocol – custom protocol or event was detected;
- protocol-command-decode – encryption attempt was detected;
- web-application-activity – access to a potentially vulnerable web application;
- web-application-attack – attack on web application;
- misc-activity – other activity;
- misc-attack – other attacks;
- icmp-event – general ICMP event;
- inappropriate-content – inappropriate content was detected;
- policy-violation – potential breach of corporate privacy;
- default-login-attempt – login attempt using a standard login/password.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# meta classification-type misc-attack
```

meta log-message

This command defines the text message that the IPS/IDS service will generate when the rule will be triggered.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
meta log-message <MESSAGE>
[no] mera log-message
```

Parameters

<MESSAGE> – text message, specified by a string of up to 128 characters.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# meta log-message «Possible SlowLorys attack»
```

payload content

This command specifies the contents of IP packets, if matched, the rule will be triggered.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
payload content <CONTENT>
[no] payload content <CONTENT>
```

Parameters

<CONTENT> – text message, specified by a string of up to 1024 characters.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# payload content «virus»
```

payload data-size

This command sets the packet content size at which the rule will be triggered.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
payload data-size <SIZE>
```

```
[no] payload data-size
```

Parameters

<SIZE> – packet content size, takes values in the range of [1..65535]

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# payload data-size 1024
```

payload data-size comparison-operator

Comparison operator for ip icmp type command. Applicable only in conjunction with this command.

The use of a negative form (no) of the command cancels the comparison.

Syntax

```
payload data-size comparison-operator { greater-than | less-than }
[no] payload data-size comparison-operator
```

Parameters

- greater-than – greater than;
- less-than – less than.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# payload data-size 1024
esr(config-ips-category-rule)# payload data-size comparison-operator less-than
```

payload depth

This command indicates how many bytes from the beginning of the packet contents will be checked by this rule. This command is used in conjunction with the payload content command only. It can be used in conjunction with the payload offset command.

The use of a negative form (no) of the command means that the entire contents of the package will be checked for exact compliance.

Syntax

```
payload depth <DEPTH>
```

```
[no] payload content depth
```

Parameters

<DEPTH> – the number of bytes from the beginning of the packet contents, takes a value in the range [1..65535].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# payload content «abc»
esr(config-ips-category-rule)# payload depth 3
```

Packets with the contents of 'abcdef', 'abc123', 'abcabcabc', etc., will fall under the rule.

payload no-case

This command points not to distinguish uppercase and lowercase letters in the description of package contents. This command is used in conjunction with the payload content command only.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
payload no-case
```

```
[no] payload content no-case
```

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# payload content «virus»
esr(config-ips-category-rule)# payload no-case
```

Packets with the contents of 'virus', 'VIRUS', 'ViRuS', etc., will fall under the rule.

payload offset

This command specifies the number of offset bytes from the beginning of the contents of the packet from which the check will begin. This command is used in conjunction with the payload content command only. It can be used in conjunction with the payload depth command.

The use of a negative form (no) of the command means that the entire contents of the package will be checked for exact compliance.

Syntax

```
payload offset <OFFSET>
```

```
[no] payload content offset
```

Parameters

<OFFSET> – the number of offset bytes from the beginning of the packet contents, takes a value in the range [1 .. 65535].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# payload content «abc»
esr(config-ips-category-rule)# payload depth 6
esr(config-ips-category-rule)# payload offset 3
```

Packets with the contents of '123abcdef', 'defabc', 'abcabcabc', etc., will fall under the rule.

protocol

The command sets name of IP for which the rule should work. The use of a negative form (no) of the command cancels the assignment.

Syntax

```
protocol { any | ip | icmp | http | tcp | udp }
```

```
[no] protocol
```

Parameters

- any – the rule will be triggered for any protocols;
- ip – the rule will be triggered for ip. You can configure additional filtering in the rule with the ip protocol-id command;

- icmp – the rule will be triggered for icmp. When this option is selected, the values of source-port and destination-port must be any. You can configure additional filtering in the rule with the ip icmp commands;
- http – the rule will be triggered for http. You can configure additional filtering in the rule with the ip http commands;
- tcp – the rule will be triggered for tcp. You can configure additional filtering in the rule with the ip tcp commands;
- udp – the rule will be triggered for udp. You can configure additional filtering in the rule with the ip udp commands;
- ftp – the rule will be triggered for ftp; You can configure additional filtering in the rule with the ip ftp commands;
- ftp-data – the rule will be triggered for ftp data field; You can configure additional filtering in the rule with the ip ftp-data commands;

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# protocol udp
```

rule

The command creates a rule and switches to CONFIG-IPS-CATEGORY-RULE configuration mode. The rules are proceeded by the device in number ascending order.

The use of a negative form (no) of the command removes a specified rule.

Syntax

```
[no] rule <ORDER>
```

Parameters

<ORDER> – rule number, takes values of [1..512].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY

Example

```
esr(config-ips-category)# rule 10
esr(config-ips-category-rule)#
```

security ips-category user-defined

This command creates a set of IPS/IDS service user rules with a specific name and switches to the configuration mode of this set.

The use of a negative form (no) of the command removes the configured policy of the IPS service settings.

Syntax

```
[no] security ips-category user-defined <CATEGORY_NAME>
```

Parameters

<CATEGORY_NAME> – name of the set of IPS/IDS service user rules, specified by a string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# security ips-category user-defined PROTOCOL
esr(config-ips-category)#
```

source-address

The command sets source IP addresses for which the rule should work.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
source-address {ip <ADDR> | ip-prefix <ADDR/LEN> | object-group <OBJ_GR_NAME> | policy-object-group { protect | external } | any }
```

```
no source-address
```

Parameters

<ADDR> – source IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<ADDR/LEN> – IP subnet, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and LEN takes values of [1..32].

<OBJ_GR_NAME> – name of IP addresses profile that contains sender IP address, set by the string of up to 31 characters.

destination-address policy-object-group protect – sets protect addresses defined in IPS/IDS policy as source addresses

destination-address policy-object-group external –sets external addresses defined in IPS/IDS policy as source addresses

When specifying the 'any' value, the rule will be triggered for any source IP address.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# source-address ip-prefix 192.168.0.0/16
```

source-port

The command sets the number of source TCP/UDP port for which the rule should work.

The use of a negative form (no) of the command removes the assignment.

Syntax

```
source-port { any | <PORT> | object-group <OBJ-GR-NAME> }
no source-port
```

Parameters

<PORT> – number of source TCP/UDP port, takes values of [1..65535].

<OBJ_GR_NAME> – sender TCP/UDP ports profile name, set by the string of up to 31 characters.

When specifying the 'any' value, the rule will be triggered for any source TCP/UDP port.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# source-port 22
```

threshold count

This command specifies the threshold number of packets at which the rule will be triggered.

The use of a negative form (no) of the command removes the assignment.

Syntax

```
threshold count <COUNT>
```

```
[no] threshold count
```

Parameters

<COUNT> – number of packets, takes values in the range of [1..65535]

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# threshold count 1024
```

threshold second

This command sets the time interval for which the threshold value is considered. packets at which the rule will be triggered. This command is used in conjunction with the threshold count command only.

The use of a negative form (no) of the command removes the assignment.

Syntax

```
threshold second <SECOND>
[no] threshold second
```

Parameters

<SECOND> – time interval in seconds, takes values in the range of [1..65535].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# threshold second 1
```

threshold track

This command sets that packets for which threshold values are set will be considered at the address of the sender or recipient. This command is used in conjunction with the threshold count command only.

The use of a negative form (no) of the command removes the assignment.

Syntax

```
threshold track { by-src | by-dst }
```

```
[no] threshold track
```

Parameters

- by-src – read threshold value for packets with the same IP sender;
- by-dst – read threshold value for packets with the same IP recipient.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# threshold track by-src
```

threshold type

This command sets the threshold processing method. This command is used in conjunction with the threshold count command only.

The use of a negative form (no) of the command removes the assignment.

Syntax

```
threshold type { treshhold | limit | both }
```

```
[no] threshold type
```

Parameters

- threshold – display a message every time a threshold is reached;
- limit – issue a message no more than <COUNT> times per time interval <SECOND>;
- both – threshold and limit combination. A message will be generated if during the <SECOND> time interval there were <COUNT> or more packets matching the rule conditions, and the message will be sent only once during the time interval;

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE

Example

```
esr(config-ips-category-rule)# threshold count 1024
esr(config-ips-category-rule)# threshold second 1
esr(config-ips-category-rule)# threshold track by-src
esr(config-ips-category-rule)# threshold type treshold
```

A message will be generated for every X*1025 packet arriving in 1 second from one IP address.

Extended user rules configuration

rule-advanced

The command creates a rule and switches to CONFIG-IPS-CATEGORY-RULE-ADVANCED configuration mode. The rules are proceeded by the device in number ascending order.

The use of a negative form (no) of the command removes a specified rule.

Syntax

```
[no] rule-advanced <ORDER>
```

Parameters

<ORDER> – rule number, takes values of [1..4294967295].

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE-ADVANCED

Example

```
esr(config-ips-category)# rule-advanced 10
esr(config-ips-category-rule-advanced)#
```

rule-text

This command describes the traffic processing rule in SNORT 2.X/Suricata 4.X format

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
rule-text <LINE>
```

```
[no] rule-text
```

Parameters

<LINE> – text message in SNORT 2.X/Suricata 4.X format, specified by a string of up to 1024 characters.

When writing rules, the symbol " needs to be replaced with the symbol '.

Required privilege level

15

Command mode

CONFIG-IPS-CATEGORY-RULE-ADVANCED

Example

```
esr(config-ips-category-rule-advanced)# rule-text «alert tcp any any -> $HOME_NET any (msg:
'ATTACK [PTsecurity] Attempt to crash named using malformed RNDC packet'; flow: established,
to_server; content:'_auth'; depth: 20; fast_pattern; content: !'|02 00 00 00|'; within: 4;
content: '_ctrl'; content: '_ser'; content: '_tim'; content: '_exp'; reference: cve, 2016-1285;
classtype: attempted-dos; reference: url, github.com/ptresearch/AttackDetection; metadata: Open
Ptsecurity.com ruleset; sid: 10000005; rev: 3; )»
```

"Antispam" system configuration

- action
- description
- description
- domain
- enable
- enable
- headers max-size
- mail domain
- mail max-size
- mail server ip
- mailserver
- mailserver domain
- mark-type
- profile antispam
- rule
- security antispam profile
- sender email
- sender ip
- smtp helo-required
- smtp vrfy-enable
- tls enable
- tls keyfile

action

This command is used to specify the action to be applied to the email on which the rule was triggered.

The use of a negative form (no) of the command removes an assigned action.

Syntax

```
action <ACTION>
[no] action
```

Parameters

<ACTION> – assigned action:

- reject – further delivery of the letter is prohibited, the sender of the letter is sent a reply about the error.

Default value

None

Required privilege level

15

Command mode

CONFIG-ANTISPAM-PROFILE-RULE

Example

```
esr(config-antispam-profile-rule)# action reject
```

description

This command is used to change the description of the customized "Antispam" service profile or the rule within the "Antispam" service profile.

The use of a negative form (no) of the command removes a specified description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – description of the configured "Antispam" service profile or rules within the "Antispam" service profile, specified by a string of up to 255 characters.

Default value

None

Required privilege level

15

Command mode

CONFIG-ANTISPAM-PROFILE

CONFIG-ANTISPAM-PROFILE-RULE

Example

```
esr(config-antispam-profile)# description "Test Antispam Profile 1"
```

description

This command is used to change the description of the configured mail domain.

The use of a negative form (no) of the command removes a specified description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

< DESCRIPTION > – description of the configured mail domain, specified by a string of up to 255 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-MAILSERVER-DOMAIN

Example

```
esr(config-mailserver-domain)# description "Test Mail Domain"
```

domain

This command is used to specify a mail domain.

The use of a negative form (no) of the command removes the specified mail domain.

Syntax

```
domain <DOMAIN-NAME>
no domain
```

Parameters

<DOMAIN-NAME> – name of the email domain, a string of up to 31 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-MAILSERVER

Example

```
esr(config-mailserver)# domain test_domain
```

enable

This command is used to activate a rule in the "Antispam" service profile.

The use of a negative form (no) of the command enables a rule.

Syntax

`[no] enable`

Parameters

None.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-ANTISPAM-PROFILE-RULE

Example

```
esr(config-antispam-profile)# enable
```

enable

This command is used to activate the mail server on the basis of which the "Antispam" service operates or the mail domain.

The use of a negative form (no) of the command deactivates mail server or main domain.

Syntax

`[no] enable`

Parameters

None.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-MAILSERVER

CONFIG-MAILSERVER-DOMAIN

Example

```
esr(config-mailserver-domain)# enable
```

headers max-size

This command is used to specify the maximum size of an email header. If the size of the header of the message to be processed exceeds the specified value, the unnecessary entries in the header will be discarded and will not be taken into account when processing the message.

The use of a negative form (no) of the command sets the default maximum email header size.

Syntax

```
headers max-size <SIZE>
no headers max-size
```

Parameters

<SIZE> – maximum size of the email header in KB, takes values in the range [50...200].

Default value

100.

Required privilege level

10

Command mode

CONFIG-MAILSERVER

Example

```
esr(config-mailserver)# headers max-size 115
```

mail domain

This command is used to specify the name of the email domain for which the "Antispam" service will receive, analyze and deliver mail according to the settings of the email domain. Emails that do not come from or for the domain you specify will be discarded.

The use of a negative form (no) of the command removes a specified mail domain name.

Syntax

```
mail domain <NAME>
[no] mail domain
```

Parameters

<NAME> – name of the email domain, a string of up to 63 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-MAILSERVER-DOMAIN

Example

```
esr(config-mailserver-domain)# mail domain eltex-co.ru
```

mail max-size

This command is used to specify the maximum size of an email. If the size of the message to be processed exceeds the specified value, the message will be discarded and the sender of the message will get a corresponding notification.

The use of a negative form (no) of the command sets the default maximum email size.

Syntax

```
mail max-size <SIZE>
no mail max-size
```

Parameters

<SIZE> – maximum size of an email in KB. Takes values in the range of [5120..51200].

Default value

15360.

Required privilege level

10

Command mode

CONFIG-MAILSERVER

Example

```
esr(config-mailserver)# mail max-size 10000
```

mail server ip

This command is used to specify the IP address of the mail server, for which the "Antispam" service will redirect email according to the settings of the mail domain. Emails coming from a domain specified in the "mail domain" command but not from the specified IP address will be rejected.

The use of a negative form (no) of the command removes a specified mail server IP address.

Syntax

```
mail server ip <ADDR>
[no] mail server ip
```

Parameters

<ADDR> – mail server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Default value

None

Required privilege level

10

Command mode

CONFIG-MAILSERVER-DOMAIN

Example

```
esr(config-mailserver-domain)# mail server ip 192.168.1.1
```

mailserver

This command is used to switch to the configuration mode of the mail server, on the basis of which the "Antispam" service operates.

The use of a negative form (no) of the command removes the mail server from the configuration.

Syntax

`[no] mailserver`

Parameters

None.

Default value

None

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# mailserver
```

mailserver domain

This command is used to create a mail domain, which is used when the mail server is running.

The use of a negative form (no) of the command removes the specified mail domain.

Syntax

`[no] mailserver domain <DOMAIN-NAME>`

Parameters

<DOMAIN-NAME> – name of the email domain, a string of up to 31 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-MAILSERVER

Example

```
esr(config)# mailserver domain local_domain
```

mark-type

This command is used to specify the marking type for emails that the "Antispam" service has classified as "Spam".

The use of a negative form (no) of the command removes the marking type for emails that the "Antispam" service classified as "Spam".

Syntax

```
mark-type <MARK-TYPE>
no mark-type
```

Parameters

<MARK-TYPE> – the marking type of an email categorized as "Spam". Takes the following values:

- header – add the X-Spam header to email headers;
- subject – add the [SPAM] tag before the subject of the email.

Default value

None

Required privilege level

15

Command mode

CONFIG-ANTISPAM-PROFILE

Example

```
esr(config-antispam-profile)# mark-type header
```

profile antispam

This command is used to specify the "Antispam" service profile to use in the email domain configuration.

The use of a negative form (no) of the command removes a specified profile from the email domain configuration.

Syntax

```
profile antispam <ANTISPAM-PROFILE-NAME>
[no] profile antispam
```

Parameters

<ANTISPAM-PROFILE-NAME> – name of the "Antispam" service profile, specified as a string of up to 31 characters.

Default value

None

Required privilege level

15

Command mode

CONFIG-MAILSERVER-DOMAIN

Example

```
esr(config-mailserver-domain)# profile antispam main_domain_protection
```

rule

This command is used to create a rule in the "Antispam" service profile and switch to its configuration mode. The rules are proceeded by the device in number ascending order.

The use of a negative form (no) of the command removes a specified rule.

Syntax

```
[no] rule <ORDER>
```

Parameters

<ORDER> – rule number, may take values [1..100].

Default value

Rule is not created.

Required privilege level

15

Command mode

CONFIG-ANTISPAM-PROFILE

Example

```
esr(config-antispam-profile)# rule 10
```

security antispam profile

This command is used to create an "Antispam" service profile and switch to its configuration mode.

The use of a negative form (no) of the command removes a specified "Antispam" service profile.

Syntax

```
[no] security antispam profile <NAME>
```

Parameters

<NAME> – name of the Antispam service profile, specified as a string of up to 31 characters.

Default value

None

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# security antispam profile TEST
```

sender email

This command is used to specify a group of domains or mailboxes from which mails are sent, for which the rule should trigger.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
sender email <OBJ-GROUP-MAIL-NAME>
```

```
[no] sender email
```

Parameters

<OBJ-GROUP-MAIL-NAME> – name of a group of domains or mailboxes, specified as a string of up to 31 characters.

Default value

None

Required privilege level

15

Command mode

CONFIG-ANTISPAM-PROFILE-RULE

Example

```
esr(config-antispam-profile-rule)# sender email mail_blacklist
```

sender ip

This command is used to specify the group of IP addresses of mail servers or clients for which the rule should trigger.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
sender ip <OBJ-GROUP-NETWORK-NAME>
```

```
[no] sender ip
```

Parameters

<OBJ-GROUP-NETWORK-NAME> – IP addresses profile name, set by the string of up to 31 characters.

Default value

None

Required privilege level

15

Command mode

CONFIG-ANTISPAM-PROFILE-RULE

Example

```
esr(config-antispam-profile-rule)# sender ip ip_blacklist
```

smtp helo-required

This command is used to enable the mandatory HELO/EHLO SMTP command from an SMTP client that tries to connect to ESR via SMTP. None of the HELO/EHLO commands during the establishment of an SMTP session will result in the connection being dropped.

The use of a negative form (no) of the command disables the mandatory requirement of the HELO/EHLO command.

Syntax

```
[no] smtp helo-required
```

Parameters

None.

Default value

The HELO/EHLO command is not required during an SMTP session establishment.

Required privilege level

10

Command mode

CONFIG-MAILSERVER

Example

```
esr(config-mailserver)# smtp helo-required
```

smtp vrfy-enable

This command is used to enable the VRFY SMTP command, which allows you to check if a user with this name is on the list of users served by this mail server. This command is disabled by default in order to protect against scanning the user base on the mail server.

The use of a negative form (no) of the command disables the possibility to use the VRFY SMTP command.

Syntax

```
[no] smtp vrfy-enable
```

Parameters

None.

Default value

The VRFY command is disabled.

Required privilege level

10

Command mode

CONFIG-MAILSERVER

Example

```
esr(config-mailserver)# smtp vrfy-enable
```

tls enable

This command is used to enable the TLS protocol to support secure connections to other mail servers and mail clients. If TLS is enabled on ESR and the other party does not report TLS support, the whole SMTP session will be in plaintext (i.e. open).

The use of a negative form (no) of the command disables TLS protocol support.

Syntax

`[no] tls enable`

Parameters

None.

Default value

TLS support is disabled.

Required privilege level

10

Command mode

CONFIG-MAILSERVER

Example

```
esr(config-mailserver)# tls enable
```

tls keyfile

This command is used to specify certificates and keys for TLS during SMTP sessions establishment.

The use of a negative form (no) of the command removes a specified certificate or key type.

Syntax

```
tls keyfile <TYPE> <NAME>
[no] tls keyfile
```

Parameters

<TYPE> – type of certificate or key file, takes values:

- ca – certificate authority;
- server-key – private server key;
- server-crt – public server certificate;
- dh – Diffie-Hellman key.

<NAME> – certificate file name, set by the string of up to 31 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-MAILSERVER

Example

```
esr(config-mailserver)# tls keyfile server-crt server.crt
```

27 QoS management

- class
- class-map
- compression header ip tcp
- description
- fair-queue
- ip firewall sessions classification enable
- match access-group
- match dscp
- mode
- policy-map
- priority class
- priority level
- priority-queue out
- priority-queue out num-of-queues
- qos dscp-mutation
- qos enable
- qos map cos-queue
- qos map dscp-mutation
- qos map dscp-queue
- qos queue default
- qos trust
- qos wrr-queue
- queue-limit
- random-detect
- random-detect queue
- random-detect queue default
- rate-limit
- service-policy
- service-policy
- set class-default cos
- set class-default dscp
- set class-default ip-precedence
- set cos
- set dscp
- set ip-precedence
- set queue
- shape auto-distribution
- shape average
- shape peak
- show qos interface shapers
- show qos map cos-queue
- show qos map dscp-mutation
- show qos map dscp-queue
- show qos policy binding
- show qos policy configuration
- show qos policy statistics
- show qos statistics
- show qos tunnel shapers
- traffic-shape

class

This command binds the specified QoS-class to the policy and the transition to the configuration mode of the class parameters.

The use of a negative form (no) of the command removes a bind of class to the policy.

Syntax

```
[no] class <NAME>
```

Parameters

<NAME> – name of the class being bound, set by the string of up to 31 characters. When specifying the 'class-default' value, the incoming unclassified traffic falls into this class.

Required privilege level

10

Command mode

CONFIG-POLICY-MAP

Example

```
esr(config-policy-map)# class telnet_traffic
esr(config-class-policy-map)#
```

class-map

Create QoS class and switch to the class parameters configuration mode.

The use of a negative form (no) of the command removes a created rule.

Syntax

```
[no] class-map <NAME>
```

Parameters

<NAME> – name of the class being created, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# class-map telnet_traffic
```

compression header ip tcp

This command enables tcp headers compression protocol for the certain class traffic.

The use of a negative form of the command (no) cancels the use of the tcp header compression protocol for traffic of a particular class.

Syntax

```
[no] compression header ip tcp
```

Parameters

None.

Required privilege level

10

Command mode

CONFIG-POLICY-MAP-CLASS

Example

```
esr(config-class-policy-map)# compression header ip tcp
```

description

This command defines QoS classificator description.

The use of a negative form (no) of the command removes description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – interface description, set by the string of up to 255 characters.

Required privilege level

15

Command mode

CONFIG-CLASS-MAP

CONFIG-POLICY-MAP

Example

```
esr(config-class-map)# description "VoIP"
```

fair-queue

Specify the limited number of virtual queues.

The use of a negative form (no) of the command sets the default value.

Syntax

```
fair-queue <QUEUE-LIMIT>
no fair-queue
```

Parameters

<QUEUE-LIMIT> – limited number of virtual queues, takes values in the range of [16..4096].

Default value

16

Required privilege level

10

Command mode

CONFIG-POLICY-MAP-CLASS

Example

```
esr(config-class-policy-map)# fair-queue 200
```

ip firewall sessions classification enable

This command enables session classification based on QoS policy.

The use of a negative form (no) of the command disables session classification.

Syntax

```
[no] ip firewall sessions classification enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip firewall sessions classification enable
```

match access-group

This command binds the access control list (ACL), which will define the ratio of incoming traffic to a configurable class.

The use of a negative form (no) of the command removes the binding of access control list to this class.

Syntax

```
[no] match access-group {<NAME> | all }
```

Parameters

<NAME> – access control list name, set by the string of up to 31 characters.

When deleting, it is possible to use the 'all' key to delete all specified access lists.

Required privilege level

10

Command mode

CONFIG-CLASS-MAP

Example

```
esr(config-if-gi)# match access-group acl-ssh-traffic
```

match dscp

This command sets the value of the DSCP code, the traffic of which will be processed in this class-map.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
[no] match dscp <DSCP>
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

When deleting, it is possible to use the 'all' key to delete all specified dscp.

Required privilege level

10

Command mode

CONFIG-CLASS-MAP

Example

```
esr(config-class-map)# match dscp 55
```

mode

This command defines mode of class operation.

The use of a negative form (no) of the command sets the default value.

Syntax

```
mode <MODE>
```

```
no mode
```

Parameters

<MODE> – class mode:

- fifo – FIFO mode (First In, First Out);
- gred – GRED mode (Generalized RED);
- red – RED mode (Random Early Detection);
- sfq – SFQ mode (SFQ queue allocates flow-based packets transmission).

Default value

FIFO

Required privilege level

10

Command mode

CONFIG-POLICY-MAP-CLASS

Example

```
esr(config-class-policy-map)# mode red
```

policy-map

Create QoS policy and switch to the policy parameters configuration mode.

The use of a negative form (no) of the command removes a created policy.

Syntax

```
[no] policy-map <NAME>
```

Parameters

<NAME> – name of the policy being created, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# policy-map input_policy
```

priority class

This command stes class priority in WRR process. Classes with the highest priority are proceeded first.

The use of a negative form (no) of the command sets the default value.

Syntax

```
priority class <PRIORITY>
no priority
```

Parameters

<PRIORITY> – priority of class in WRR process, takes values of [1..8].

Default value

1

Required privilege level

10

Command mode

CONFIG-POLICY-MAP-CLASS

Example

```
esr(config-class-policy-map)# priority class 5
```

priority level

This command switchs the class to Strict Priority mode and sets class priority. Classes with the highest priority are proceeded first.

The use of a negative form (no) of the command disables the WRR mode.

Syntax

```
priority level <PRIORITY>
no priority
```

Parameters

<PRIORITY> – priority of class in WRR process, takes values of [1..8].

Default value

The default value: tha class operates in WRR mode, the priority is not specified.

Required privilege level

10

Command mode

CONFIG-POLICY-MAP-CLASS

Example

```
esr(config-class-policy-map)# priority level 5
```

priority-queue out

This command sets limit of packets in queue.

The use of a negative form (no) of the command sets the default value.

Syntax

```
priority-queue out <QUEUE> limit <LIMIT>
no priority-queue out <QUEUE> limit
```

Parameters

<QUEUE> – queue identifier, takes values in the range of [1..8];

<QUEUE-LIMIT> – limited number of packets in a virtual queue, takes values in the range of [100..-1000].

Default value

500

Required privilege level

10

Command mode

CONFIG-GI [1]

CONFIG-SUBIF

CONFIG-PORT-CHANNEL [1]

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TPV3

Example

```
esr(config-if)# priority-queue out 1 limit 800
```

[1] This functionality is supported on the ESR-10/12V/12VF/14VF/20/21/100/200 routers only

priority-queue out num-of-queues

This command sets the number of priority queues from the eighth to the first. The remaining queues are weighted.

The use of a negative form (no) of the command sets the default value.

Syntax

```
priority-queue out num-of-queues <VALUE>
```

```
no priority-queue out num-of-queues
```

Parameters

<VALUE> – amount of queues, takes values of [1..8], where:

- 0 – all queues take part in WRR (WRR – weight-based queue processing mechanism);
- 8 – all queues are served as «strictpriority» (strictpriority – priority queue is served as soon as the packets appear).

Default value

8

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# priority-queue out num-of-queues 5
```

qos dscp-mutation

Enable DSCP codes changes according to the DSCP-Mutation table. DSCP codes are changed only for inbound traffic of trusted ports in QoS Basic mode.

The use of a negative form (no) of the command disables DSCP codes changing.

Syntax

```
[no] qos dscp-mutation
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# qos wrr-queue 3 bandwidth 130
```

qos enable

The command enables QoS on the interface. If a QoS policy is not associated with an interface (the policy binding is described in section service-policy), then the interface operates in Basic QoS mode, otherwise Policy-based QoS.

Basic QoS – traffic classification is performed based on DSCP and/or 802.1p codes depending on the selected trust mode (the command is described in section qos trust). Traffic is sent to the queues in accordance with the DSCP-Queue and/or CoS-Queue tables.

Policy-based QoS — classification and direction of traffic in the queue is based on QoS policies. Each policy defines a set of classes into which traffic should be divided. The ratio of traffic to a specific class of policy is defined at the entrance to the router by ACL rules (the ACL binding is described in section match access-group), for this purpose, a QoS policy is assigned to the incoming direction. To limit the bandwidth of previously classified traffic and other functions, a QoS policy is assigned to the outbound direction.

The use of a negative form (no) of the command disables QoS service on the interface.

Syntax

```
[no] qos enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-E1

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-VTI

CONFIG-L2TPV3

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-L2TP

CONFIG-OPENVPN

Example

```
esr(config-if-gi)# qos enable
```

qos map cos-queue

Set the match between 802.1p codes values of incoming packets and outgoing queues.

The use of a negative form (no) of the command sets the default value.

Syntax

```
qos map cos-queue <COS> to <QUEUE>
no qos map dscp-queue <COS>
```

Parameters

<COS> – service classifier in 802.1q packet tag, takes values in the range of [0..7];

<QUEUE> – queue identifier, takes values in the range of [1..8].

Default values:

CoS: (0), queue 1

CoS: (1), queue 2

CoS: (2), queue 3

CoS: (3), queue 4

CoS: (4), queue 5

CoS: (5), queue 6

CoS: (6), queue 7

CoS: (7), queue 8

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# qos map cos-queue 7 to 5
```

qos map dscp-mutation

Set the match between DSCP codes values of incoming packets and outgoing DSCP codes.

The use of a negative form (no) of the command sets the default value.

Syntax

```
qos map dscp-mutation <DSCP> to <DSCP>
no qos map dscp-mutation <DSCP>
```

Parameters

<DSCP> – service classifier in a packet IP header, takes values in the range of [0..63].

Default values:

Set the match between DSCP codes values of incoming packets and outgoing DSCP codes.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# qos map dscp-mutation 10 to 40
```

qos map dscp-queue

Set the match between DSCP codes values of incoming packets and outgoing queues.

The use of a negative form (no) of the command sets the default value.

Syntax

```
qos map dscp-queue <DSCP> to <QUEUE>
no qos map dscp-queue <DSCP>
```

Parameters

<DSCP> – service classifier in a packet IP header, takes values in the range of [0..63];

<QUEUE> – internal queue to which packets with the previously described DSCP must be sent takes the value [1..8].

Default values:

DSCP: (0-7), queue 1

DSCP: (8-15), queue 2

DSCP: (16-23), queue 3

DSCP: (24-31), queue 4

DSCP: (32-39), queue 5

DSCP: (40-47), queue 6

DSCP: (48-55), queue 7

DSCP: (56-63), queue 8

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# qos map dscp-queue 42 to 5
```

qos queue default

This command sets the default queue number for all traffic except IP in DSCP priorities trust mode in the case of Basic QoS, as well as non-classified traffic in the case of Policy-based QoS.

The use of a negative form (no) of the command sets the default value.

Syntax

```
qos queue default <QUEUE>
no qos queue default
```

Parameters

<QUEUE> – queue identifier, takes values in the range of [1..8].

Default value

1

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# qos queue default 3
```

qos trust

This command sets the mode of trusting the values of 802.1p and DSCP codes in incoming packets for the Basic QoS mode of the interface.

The use of a negative form (no) of the command sets the default value.

Syntax

```
qos trust <MODE>
no qos trust
```

Parameters

<MODE> – trust mode for 802.1p and DSCP codes values, takes one of the following values:

- dscp – trust mode for DSCP codes values in IP header. Not IP packets will be sent to the default queue (the command is described in section qos queue default);
- cos – trust mode for 802.1p codes values in 802.1q tag. Untagged packets will be sent to the default queue (the command is described in section qos queue default);
- cos-dscp – trust mode for DSCP codes values in IP packets and for 802.1p codes values in other packets.

Default value

dscp – trust mode for DSCP codes values in IP header.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# qos trust cos-dscp
```

qos wrr-queue

This command defines weights of corresponded weighted queues.

The use of a negative form (no) of the command sets the default queue weight value.

Syntax

```
qos wrr-queue <QUEUE> bandwidth <WEIGHT>
```

```
no qos wrr-queue <QUEUE>
```

Parameters

<QUEUE> – queue identifier, takes values in the range of [1..8];

<WEIGHT> – weight value, takes values in the range of [1..255].

Default value

1

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# qos wrr-queue 3 bandwidth 130
```

queue-limit

Specify the limited number of packets for a virtual queue.

The use of a negative form (no) of the command sets the default value.

Syntax

```
queue-limit <QUEUE-LIMIT>
no queue-limit
```

Parameters

<QUEUE-LIMIT> – limited number of packets in a virtual queue, takes values in the range of [2..4096].

Default value

127

Required privilege level

10

Command mode

CONFIG-POLICY-MAP-CLASS

Example

```
esr(config-class-policy-map)# queue-limit 200
```

random-detect

This command defines RED parameters.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
random-detect <LIMIT> <MIN> <MAX> <APS> <APS-NUM> <PROBABILITY>
no random-detect
```

Parameters

<LIMIT> – limited size of a queue in bytes, takes values of in the range of [1..1000000];

<MIN> – minimum size of a queue in bytes, takes value in the range of [1..1000000];

<MAX> – maximum size of a queue in bytes, takes value in the range of [1..1000000];

<MIN> – minimum size of a queue in bytes, takes value in the range of [1..10000000];

<APS-NUM> – the number of medium-size packets allowed for short-term transmission, takes a value in the range [0..10000000];

<PROBABILITY> – probability of packet drop, takes values of [0..100].

When specifying values, the following rules should be followed:

<MAX> > 2 * <MIN> <LIMIT> >3 * <MAX>

Required privilege level

10

Command mode

CONFIG-POLICY-MAP-CLASS

Example

```
esr(config-class-policy-map)# random-detect 9000 1500 3000 10
```

random-detect queue

This command defines GRED parameters.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
random-detect queue <QUEUE-NUM> [ dscp <DSCP> | precedence <IPP> ] <LIMIT> <MIN> <MAX>
<APS> <APS-NUM> <PROBABILITY>
```

```
no random-detect queue <PRECEDENCE>
```

Parameters

<QUEUE-NUM> – queue number [1..16];

<DSCP> – service classifier in a packet IP header, takes values in the range of [0..63];

<IPP> – IP Precedence code value, takes values in the range of [0..7];

<PRECEDENCE> – IPPrecendence value [0..7];

<LIMIT> – limited size of a queue in bytes, takes values of in the range of [1..1000000];

<MIN> – minimum size of a queue in bytes, takes value in the range of [1..1000000];

<MAX> – maximum size of a queue in bytes, takes value in the range of [1..1000000];

<MIN> – minimum size of a queue in bytes, takes value in the range of [1..10000000];

<APS-NUM> – the number of medium-size packets allowed for short-term transmission, takes a value in the range [0..10000000];

<PROBABILITY> – probability of packet drop, takes values of [0..100].

When specifying values, the following rules should be followed:

<MAX> > 2 * <MIN>

<LIMIT> > 3 * <MAX>

Required privilege level

10

Command mode

CONFIG-POLICY-MAP-CLASS

Example

```
esr(config-class-policy-map)# random-detect precedence 2 9000 1500 3000 10
```

random-detect queue default

This command defines default queue for GRED mechanism.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
random-detect queue default <QUEUE-NUM>
no random-detect queue default <QUEUE-NUM>
```

Parameters

<QUEUE-NUM> – queue number [1..16];

Required privilege level

10

Command mode

CONFIG-POLICY-MAP-CLASS

Example

```
esr(config-class-policy-map)# random-detect queue default
```

rate-limit

This command limits the incoming traffic speed. The command is relevant only for BasicQoS mode of the interface.

The use of a negative form (no) of the command removes the restriction.

Syntax

```
rate-limit <BANDWIDTH> [BURST]
```

```
no rate-limit
```

Parameters

<BANDWIDTH> – average traffic rate in Kbps, takes the value of [1..10000000] for TengigabitEthernet interfaces and [1..1000000] for other interfaces and tunnels;

<BURST> – size of the restrictive threshold in KB, takes the value [4..16000]. By default 128 KB.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-E1

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-L2TPV3

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-OPENVPN

Example

```
esr(config-if-gi)# rate-limit 1670000
```

service-policy

This command binds the specified QoS policy to a configurable interface to classify incoming (input) or prioritize outgoing (output) traffic.

The use of a negative form (no) of the command removes a bind of policy to this interface.

Syntax

```
service-policy { input | output } <NAME>
service-policy dynamic { upstream | downstream | all }
no service-policy { input | output }
```

Parameters

<NAME> – QoS policy name, set by the string of up to 31 characters;

dynamic — binds a dynamic QoS policy to a configurable interface for prioritizing upstream traffic, downstream traffic, or all traffic. The policy is populated with data received from the RADIUS server.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-E1

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-L2TPV3

CONFIG-SUBTUNNEL (With "input" key only)

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-L2TP

CONFIG-OPENVPN

Example

```
esr(config-if-gi)# service-policy input input_policy
```

service-policy

This command binds QoS policy to class for hierarchical QoS creation.

The use of a negative form (no) of the command removes a bind of policy to the class.

Syntax

```
[no] service-policy <NAME>
```

Parameters

<NAME> – policy name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-POLICY-MAP-CLASS

Example

```
esr(config-class-policy-map)# service-policy input_policy
```

set class-default cos

Specify 802.1p priority value which will be set in packets corresponding to the default class (class-default).

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
set class-default cos <COS>
no set class-default cos
```

Parameters

<COS> – service classifier in 802.1q packet tag, takes values in the range of [0..7].

Required privilege level

10

Command mode

CONFIG-POLICY-MAP

Example

```
esr(config-class-policy-map)# set class-default cos 5
```

set class-default dscp

Specify DSCP code value which will be set in IP packets corresponding to the default class (class-default).

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
set class-default dscp <DSCP>
no set class-default dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Required privilege level

10

Command mode

CONFIG-POLICY-MAP

Example

```
esr(config-class-map)# set class-default dscp 16
```

set class-default ip-precedence

Specify IP Precedence code value which will be set in IP packets corresponding to the default class (class-default).

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
set class-default ip-precedence <IPP>
```

```
no set class-default ip-precedence
```

Parameters

<IPP> – IP Precedence code value, takes values in the range of [0..7].

Required privilege level

10

Command mode

CONFIG-POLICY-MAP

Example

```
esr(config-class-map)# set class-default ip-precedence 5
```

set cos

Specify 802.1p code value which will be set in packets corresponding to the class being configured.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
set cos <COS>
no set cos
```

Parameters

<COS> – service classifier in 802.1q packet tag, takes values in the range of [0..7].

Required privilege level

10

Command mode

CONFIG-CLASS-MAP

Example

```
esr(config-class-policy-map)# set cos 5
```

set dscp

Specify DSCP code value which will be set in IP packets corresponding to the class being configured.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
set dscp <DSCP>
no set dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Required privilege level

10

Command mode

CONFIG-CLASS-MAP

Example

```
esr(config-class-map)# set dscp 16
```

set ip-precedence

Specify IP Precedence code value which will be set in IP packets corresponding to the class being configured.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
set ip-precedence <IPP>
no set ip-precedence
```

Parameters

<IPP> – IP Precedence code value, takes values in the range of [0..7].

Required privilege level

10

Command mode

CONFIG-CLASS-MAP

Example

```
esr(config-class-map)# set ip-precedence 5
```

set queue

⚠ In the current firmware version this functionality is supported only by ESR-1000/1200/1500/1700 routers

This command sets the number of the output QoS queue to which the packets corresponding to the configurable class will be sent.

The use of a negative form (no) of the command cancels the assignment.

Syntax

```
set queue <QUEUE>
```

```
no set queue
```

Parameters

<QUEUE> – number of the output QoS hardware queue, takes the values [1..8].

Required privilege level

10

Command mode

CONFIG-CLASS-MAP

Example

```
esr(config-class-map)# set queue 5
```

shape auto-distribution

Enable automatic bandwidth allocation between classes without bandwidth configuration, including the default class.

The use of a negative form (no) of the command disables automatic band allocation.

Syntax

```
[no] shape auto-distribution
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-POLICY-MAP

Example

```
esr(config-policy-map)# shape auto-distribution
```

shape average

This command sets the guaranteed outgoing traffic bandwidth for a specific class or policy as a whole.

The use of a negative form (no) of the command removes the restriction.

Syntax

```
shape average <BANDWIDTH> [<BURST>]
no shape average
```

Parameters

<BANDWIDTH> – committed bandwidth in Kbps, takes the value of [1..10000000];

<BURST> – size of the restrictive threshold in bytes, takes the value:

- for ESR-10/12V/12VF/14VF/20/21/100/200/1000 – [128..16000000];
- for ESR-1200/1500 – [1000..4096000].

Default values:

<BANDWIDTH> – none

<BURST> – 128000

Required privilege level

10

Command mode

CONFIG-POLICY-MAP

CONFIG-POLICY-MAP-CLASS

Example

```
esr(config-policy-map)# shape average 100000 2000
```

shape peak

Set the shared outgoing bandwidth for a specific class. The class may occupy the bandwidth if a lower priority class has not occupied its committed bandwidth. The configuration is only relevant for weighted mode classes (WRR).

The use of a negative form (no) of the command removes the restriction.

Syntax

```
shape peak <BANDWIDTH> [BURST]
```

```
no shape peak
```

Parameters

<BANDWIDTH> – restrictive bandwidth in Kbps, takes the value [1..10000000];

<BURST> – size of the restrictive threshold in bytes, takes the value:

- for ESR-10/12V/12VF/14VF/20/21/100/200/1000 – [128..16000000];
- for ESR-1200/1500 – [1000..4096000].

Default values:

<BANDWIDTH> – none

<BURST> – 128000

Required privilege level

10

Command mode

CONFIG-POLICY-MAP-CLASS

Example

```
esr(config-policy-map)# shape average 100000 2000
```

show qos interface shapers

This command displays QoS parameters of the network interfaces.

Syntax

```
show qos interface shapers <IF>
```

Parameters

<IF> – an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show qos interface shapers gigabitethernet 1/0/2
gigabitethernet 1/0/2
Committed rate:  100000 Kbps
Committed burst: 1600 KBytes
```

show qos map cos-queue

This command displays information about the compliance of the 802.1p codes in the packets and output queues used in QoS.

Syntax

`show qos map cos-queue`

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show qos map dscp-queue
d1 : d2   0  1  2  3  4  5  6  7
-------   -------------------------
0         01 02 03 04 05 06 07 08
```

show qos map dscp-mutation

This command displays information about the correspondence of DSCP codes in batches and DSCP codes after changes.

Syntax

```
show qos map dscp-mutation
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show qos map dscp-mutation
d1 : d2   0  1  2  3  4  5  6  7  8  9
-------   -------------------------------
0         00 01 02 03 04 05 06 07 08 09
1         10 11 12 13 14 15 16 17 18 19
2         20 21 22 23 24 25 26 27 28 29
3         30 31 32 33 34 35 36 37 38 39
4         40 41 42 43 44 45 46 47 48 49
5         50 51 52 53 54 55 56 57 58 59
6         60 61 62 63
```

show qos map dscp-queue

This command displays information about the compliance of the DSCP codes in the packets and output queues used in QoS.

Syntax

```
show qos map dscp-queue
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show qos map dscp-queue
d1 : d2   0  1  2  3  4  5  6  7  8  9
-------   -------------------------------
0         01 01 01 01 01 01 01 01 02 02
1         02 02 02 02 02 02 03 03 03 03
2         03 03 03 03 04 04 04 04 04 04
3         04 04 05 05 05 05 05 05 05 05
4         06 06 06 06 06 06 06 06 07 07
5         07 07 07 07 07 07 08 08 08 08
6         08 08 08 08
```

show qos policy binding

This command displays the binded QoS policy to specified interface to classify incoming (input) or prioritize outgoing (output) traffic.

Syntax

```
show qos policy binding [ <IF> | <TUN> ]
```

Parameters

<IF> – an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show qos policy binding gi1/0/1
gigabitethernet 1/0/1
Output: parent
    Class: class1
        Policy: child1
            Class: class3
    Class: class2
        Policy: child2
```

show qos policy configuration

This command displays configuration of specified QoS polocy.

Syntax

```
show qos policy configuration <NAME>
```

Parameters

<NAME> – policy name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show qos policy configuration parent
Policy-map: parent
    Shape average:
        Bandwidth:          --
        Burst:              128
    Class: class1
        Set:
            COS:            --
            DSCP:           --
            IP-Precedence:  --
            Queue:          --
        Match access group: --
        Shape average:
            Bandwidth:      --
            Burst:          128
        Shape peak:
            Bandwidth:      --
            Burst:          128
        Mode:               FIFO
        Priority class:     8
        Queue limit:        127
        Service policy:     child1
    Class: class2
        Set:
            COS:            --
            DSCP:           --
            IP-Precedence:  --
            Queue:          --
        Match access group: --
        Shape average:
            Bandwidth:      --
            Burst:          128
        Shape peak:
            Bandwidth:      --
            Burst:          128
        Mode:               FIFO
        Priority class:     8
        Queue limit:        127
        Service policy:     child2
```

show qos policy statistics

This command displays statistics on sent and dropped packets. The command is relevant only for BasicQoS mode of the interface.

Syntax

```
show qos policy statistics [ <IF> | <TUN> ]
```

Parameters

<IF> – an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

Required privilege level

1

Command mode

ROOT

Example

```
esr#  sh qos policy statistics
gigabitethernet 1/0/2
   Policy pom-1
   Input policy root
   Input class root
   Shape: bytes 750947679, packets 496667, drops 1002200
       Class clm-1
           Shape: bytes 750946896, packets 496658, drops 1002200
       Class class-default
           Shape: bytes 783, packets 9, drops 0
```

show qos statistics

This command displays statistics on sent and dropped packets. The command is relevant only for BasicQoS mode of the interface.

Syntax

```
show qos statistics [ <IF> | <TUN> ]
```

Parameters

<IF> – an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show qos statistics vti 2
vti 2
Queue   Bytes                  Packets                Drops
-----   --------------------   --------------------   --------------------
1       0                      0                      0
2       0                      0                      0
3       0                      0                      0
4       0                      0                      0
5       0                      0                      0
6       964073836      1413598          0
7       121389180      177990            1235497
8       0                      0                      0
```

show qos tunnel shapers

This command displays QoS parameters of the tunnels.

Syntax

```
show qos tunnel shapers <TUN>
```

Parameters

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show qos tunnel shapers vti 2
vti 2
qid   Target        Target
      Committed     Committed
      Rate [Kbps]   Burst [KBytes]
---   -----------   --------------
1     10000         128
2     6000          128
```

traffic-shape

Set the outgoing traffic rate limiting for a certain queue or interface in total. The command is relevant only for BasicQoS mode of the interface.

The use of a negative form (no) of the command removes the restriction.

Syntax

```
traffic-shape { <BANDWIDTH> [BURST] | queue <QUEUE> <BANDWIDTH> [BURST] }
no traffic-shape [ queue <QUEUE> ]
```

Parameters

<QUEUE> – queue identifier, takes values in the range of [1..8];

<BANDWIDTH> – average traffic rate in Kbps, may take values:

- for gigabitethernet, loopback, e1 interfaces: [1..10000000];
- for tengigabitethernet interfaces: [3000..100000000];
- for fortyengigabitethernet interfaces: [3000..40000000];
- for loopback interfaces:

<BURST> – size of the restrictive threshold in KB, takes the value [4..16000]. It is only possible to specify a value that is a multiple of 4.

Default values:

<BANDWIDTH> – none

<BURST> – 128

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-E1

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-L2TPV3

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-L2TP

CONFIG-OPENVPN

Example

```
esr(config-if)# traffic-shape queue 3 100000 2000
```

28 Monitoring and management

- Netflow management
- sFlow management
- SNMP management
- SYSLOG management
- SSH, Telnet acces configuration
- LLDP configuration
- Mirroring configuration
- Zabbix-agent and zabbix-proxy configuration

Netflow management

- ip netflow export
- netflow active-timeout
- netflow collector
- netflow domain-id
- netflow enable
- netflow export-events web-proxy
- netflow export-options
- netflow inactive-timeout
- netflow max-flows
- netflow refresh-rate
- netflow version
- port
- show netflow configuration
- show netflow statistics
- show netflow statistics cpu
- source-address

ip netflow export

This command enables the export of Netflow statistics on a network interface.

The use of a negative form (no) of the command disables the export of Netflow statistics on a network interface.

Syntax

```
[no] ip netflow export
```

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-E1

CONFIG-MULTILINK

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-BRIDGE

CONFIG-LT

CONFIG-PPPOE

Example

```
esr(config-if-gi)# ip netflow export
```

netflow active-timeout

This command sets the interval after which information about active sessions is exported to the collector.

The use of a negative form (no) of the command sets the default value.

Syntax

```
netflow active-timeout <TIMEOUT>
no netflow active-timeout
```

Parameters

<TIMEOUT> – delay before sending information about active sessions, set in seconds, takes the value [5..36000].

Default value

1800 seconds

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# netflow active-timeout 30
```

netflow collector

The command is used to set partnership relations between NTP servers and to switch to CONFIG-NTP command mode.

The use of a negative form (no) of the command removes a specific carrier information.

Syntax

```
[no] netflow collector <ADDR> [ vrf <VRF> ]
```

Parameters

<ADDR> – collector IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# netflow collector 10.100.100.1
esr(config-netflow-host)#
```

netflow domain-id

This command sets Observation Domain IP value in Netflow packets.

The use of a negative form (no) of the command sets the default value.

Syntax

```
netflow domain-id <ID>
no netflow domain-id
```

Parameters

<ID> – Observation Domain ID value, takes the value [0..4294967295].

Default value

0

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# netflow domain-id 35
```

netflow enable

This command enables Netflow on the router.

The use of a negative form (no) of the command disables Netflow on the router.

Syntax

```
[no] netflow enable
```

Parameters

The command does not contain parameters.

Default value

Process disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# netflow enable
```

netflow export-events web-proxy

This command enables transmission of netflow statistics on packets transmitted via web proxy.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] netflow export-events web-proxy
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# netflow export-events web-proxy
```

netflow export-options

This command enables additional fields in the information exported to the collector.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] netflow export-options { http-host | http-url }
```

Parameters

http-host – for HTTP/HTTPS sessions, the domain information of the host to which the HTTP request was sent will be added. The field in the Netflow packet has the identifier 24884, 128 bytes.

http-url – for HTTP sessions information about the URL to which the HTTP request was sent will be added. The field in the Netflow packet has the identifier 24887, 384 bytes.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# netflow export-options http-host
```

netflow inactive-timeout

This command sets the interval after which information about outdated sessions is exported to the collector.

The use of a negative form (no) of the command sets the default value.

Syntax

```
netflow inactive-timeout <TIMEOUT>
no netflow inactive-timeout
```

Parameters

<TIMEOUT> – delay before sending information about outdated sessions, set in seconds, takes the value [0..240].

Default value

15

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# netflow inactive-timeout 30
```

netflow max-flows

This command sets the maximum number of watched sessions.

The use of a negative form (no) of the command sets the default value.

Syntax

```
netflow max-flows <COUNT>
no netflow max-flows
```

Parameters

<COUNT> – number of watched sessions, takes the value [10000..2000000].

Default value

512000

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# netflow max-flows 300000
```

netflow refresh-rate

This command sets the frequency of sending templates to the Netflow collector.

The use of a negative form (no) of the command sets the default value.

Syntax

```
netflow refresh-rate <RATE>
no netflow refresh-rate
```

Parameters

<RATE> – frequency of sending statistics, is set in packets per stream, takes the value [1..10000].

Default value

10

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# netflow refresh-rate 100
```

netflow version

The command sets Netflow protocol version.

The use of a negative form (no) of the command sets the default value.

Syntax

```
netflow version <VERSION>
no netflow version
```

Parameters

<VERSION> – Netflow protocol version: 5, 9 and 10.

Default value

9

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# netflow version 10
```

port

This command defines the port of the Netflow service on the statistics collection server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
port <PORT>
no port
```

Parameters

<PORT> – UDP port number, set in the range of [1..65535].

Default value

2055

Required privilege level

15

Command mode

CONFIG-NETFLOW-HOST

Example

```
esr(config-netflow-host)# port 5555
```

show netflow configuration

The command displays Netflow agent configuration parameters.

Syntax

```
show netflow configuration
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show netflow configuration
Netflow configuration:
Global state:      Enabled
Version:           9
Maxflows:          10001
Refresh rate:      10
Inactive timeout:  15
Host: 115.0.0.10  Port: 2055
```

show netflow statistics

This command displays current information about the operation of Netflow.

Syntax

```
show netflow statistics
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show netflow statistics
Flows: active 9 (peak 34 reached 1d4h20m ago), mem 3841K
Hash: size 491496 (mem 3839K). InHash: 760 pkt, 339 K, InPDU 4, 160.
Processed rate   Bits/s           Packets/s
--------------   --------------   ---------------
Current                    5142                 2
1 Min Avg                  4921                 0
5 Min Avg                  4874                 0
Export: Rate 0 bytes/s; Total 3952 pkts, 3 MB, 28818 flows; Errors 2 pkts; Traffic lost 0 pkts,
0 Kbytes, 0 flows.
```

show netflow statistics cpu

This command displays statistics on the distribution of Netflow load information on the CPU.

Syntax

```
show netflow statistics
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show netflow statistics cpu
                Traffic,      Traffic,  Drop,         Drop,
cpu    PPS      Packets       MBytes    Packets       KBytes
-----  -------  ------------  --------  ------------  --------
Total        1        215224        61             0         0
cpu0         0             0         0             0         0
cpu1         0         10485         0             0         0
cpu2         0          2676         0             0         0
cpu3         0         12893         0             0         0
cpu4         0             0         0             0         0
cpu5         1        106264        53             0         0
cpu6         0          2684         0             0         0
cpu7         0         10213         0             0         0
cpu8         0          6770         0             0         0
cpu9         0          5424         0             0         0
cpu10        0          2505         0             0         0
cpu11        0         10919         1             0         0
cpu12        0         13395         0             0         0
cpu13        0          2769         0             0         0
cpu14        0         14050         0             0         0
cpu15        0         14177         1             0         0
```

source-address

This command defines the IP address of the router that will be used as the source IP address in packets sent by Netflow.

The use of a negative form (no) of the command removes a specified source IP address.

Syntax

```
source-address <ADDR>
```

```
no source-address
```

Parameters

<ADDR> – source IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

Required privilege level

15

Command mode

CONFIG-NETFLOW-HOST

Example

```
esr(config-netflow-host)# source-address 10.100.100.2
```

sFlow management

- ip sflow export
- port
- sflow collector
- sflow enable
- sflow poll-interval
- sflow sampling-rate
- show sflow configuration

ip sflow export

This command enables the export of sFlow statistics on a network interface. The sFlow function on the network interface can be enabled if the Firewall function (section ip firewall disable) is disabled on the interface; otherwise, the export of sFlow statistics is configured in the Firewall rule (section action).

The use of a negative form (no) of the command disables the export of sFlow statistics on a network interface.

Syntax

```
[no] ip sflow export
```

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-SERIAL

CONFIG-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-E1

CONFIG-MULTILINK

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-BRIDGE

CONFIG-LT

CONFIG-PPPOE

Example

```
esr(config-if-gi)# ip sflow export
```

port

This command defines the port of the sFlow service on the statistics collection server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
port <PORT>
no port
```

Parameters

<PORT> – UDP port number, set in the range of [1..65535].

Default value

6343

Required privilege level

15

Command mode

CONFIG-SFLOW-HOST

Example

```
esr(config-sflow-host)# port 5556
```

sflow collector

The command is used to set partnership relations between NTP servers and to switch to CONFIG-NTP command mode.

The use of a negative form (no) of the command removes a specific carrier information.

Syntax

```
[no] sflow collector <ADDR>
```

Parameters

<ADDR> – collector IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# sflow collector 10.100.100.1
esr(config-sflow-host)#
```

sflow enable

This command enables sFlow on the router.

The use of a negative form (no) of the command disables sFlow on the router.

Syntax

```
[no] sflow enable
```

Parameters

The command does not contain parameters.

Default value

Process disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# sflow enable
```

sflow poll-interval

This command sets the interval after which the information about the counters of the network interface is obtained.

The use of a negative form (no) of the command sets the default value.

Syntax

```
sflow poll-interval <TIMEOUT>
no sflow poll-interval
```

Parameters

<TIMEOUT> – interval after which the information about the network interface counters is retrieved takes the value [1..300].

Default value

10 seconds

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# sflow poll-interval 30
```

sflow sampling-rate

Set the rate of sending the unchanged user traffic packets to sFlow collector.

The use of a negative form (no) of the command sets the default value.

Syntax

```
sflow sampling-rate <RATE>
no sflow sampling-rate
```

Parameters

<RATE> – rate of sending the user traffic packets to the collector, takes the value of [1..65535]. If the frequency value is 10, one packet out of ten will be sent to the collector.

Default value

1000

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# sflow sampling-rate 100
```

show sflow configuration

The command displays sFlow agent configuration parameters.

Syntax

```
show sflow configuration
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show sflow configuration
sFlow configuration:
Global state:   Enabled
Sample rate:    1000
Poll interval:  10
Host: 115.0.0.10  Port: 6800
Host: 115.0.0.20  Port: 6343
Host: 115.0.0.30  Port: 6343
```

SNMP management

- access
- authentication access
- authentication algorithm
- authentication key
- client-list
- community
- ip address
- ipv6 address
- enable
- oid-tree
- port
- privacy algorithm
- privacy key
- rmon collection statistics
- snmp-server
- snmp-server community
- snmp-server contact
- snmp-server dscp
- snmp-server enable traps
- snmp-server enable traps config
- snmp-server enable traps entity
- snmp-server enable traps entity-sensor
- snmp-server enable traps environment
- snmp-server enable traps envmon
- snmp-server enable traps files-operations
- snmp-server enable traps flash
- snmp-server enable traps interfaces
- snmp-server enable traps ports
- snmp-server enable traps screens
- snmp-server enable traps snmp
- snmp-server enable traps syslog
- snmp-server enable traps wifi
- snmp-server host
- snmp-server location
- snmp-server system-shutdown
- snmp-server trap link
- snmp-server user
- snmp-server view
- source-address
- source-interface
- view

access

This command defines access level using the SNMPv3 protocol.

The use of a negative form (no) of the command sets the default value.

Syntax

```
access <TYPE>
```

```
no access
```

Parameters

<TYPE> – access level:

- ro – read only;
- rw – read/write.

Required privilege level

15

Command mode

CONFIG-SNMP-USER

Example

```
esr(config-snmp-user)# access rw
```

authentication access

This command defines security mode.

The use of a negative form (no) of the command disables the authentication.

Syntax

```
authentication access <TYPE>
```

```
no authentication access
```

Parameters

<TYPE> – security mode:

- auth – only authentication is used;
- priv – authentication and data encryption are used.

Required privilege level

15

Command mode

CONFIG-SNMP-USER

Example

```
esr(config-snmp-user)# authentication algorithm auth
```

authentication algorithm

This command defines SNMPv3 requests authentication algorithm.

The use of a negative form (no) of the command disables the authentication.

Syntax

```
authentication algorithm <ALGORITHM>
```

```
no authentication algorithm
```

Parameters

<ALGORITHM> – encryption algorithm:

- md5 – password is encrypted by md5 algorithm.
- sha1 – password is encrypted by sha1 algorithm.

Required privilege level

15

Command mode

CONFIG-SNMP-USER

Example

```
esr(config-snmp-user)# authentication algorithm md5
```

authentication key

This command sets a password for SNMPv3 requests authentication.

The use of a negative form (no) of the command removes the password.

Syntax

```
authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no authentication key
```

Parameters

<CLEAR-TEXT> – password, sets by string from 8 to 16 characters;

encrypted – when specifying a command, an encrypted password is set:

<ENCRYPTED-TEXT> – encrypted password from 8 bytes to 16 bytes (16 to 32 characters) in hexadecimal format (0xYYYY...) or (YYYY...).

Required privilege level

15

Command mode

CONFIG-SNMP-USER

Example

```
esr(config-snmp-user)# authentication key ascii-text 123456789
esr(config-snmp-user)# authentication key ascii-text encrypted CDE65039E5591FA3F1
```

client-list

This command enables filtering and sets up a profile of IP addresses from which SNMPv3 packets with this user SNMPv3 name can be received.

The use of a negative forn (no) of the command disables filtering of received SNMPv3 packets.

Syntax

`[no] client-list <NAME>`

Parameters

<NAME> – name of the previously conscious object-group, specified in a string of up to 31 characters.

Default value

Restrictions disabled.

Required privilege level

15

Command mode

CONFIG-SNMP-USER

Example

```
esr(config-snmp-user)# client-list OBG005
```

community

This command defines the SNMP community to send notifications to the remote server.

The use of a negative form (no) of the command removes the community value.

Syntax

`community <COMMUNITY>`

`no community`

Parameters

<COMMUNITY> – community for access via SNMP, set by a string [1..128] characters long;

Default value

The description is not specified.

Required privilege level

15

Command mode

CONFIG-SNMP-HOST

Example

```
esr(config-snmp-host)# community privatekey
```

ip address

This command enables the filtering and sets the IP address that is given access to the router under this SNMPv3 user.

The use of a negative forn (no) of the command disables filtering of received SNMPv3 packets.

Syntax

```
[no] ip address <ADDR>
```

Parameters

<ADDR> – IP address of client that have access, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

Default value

Restrictions disabled.

Required privilege level

15

Command mode

CONFIG-SNMP-USER

Example

```
esr(config-snmp-user)# ip address 192.168.85.33
```

ipv6 address

under this SNMPv3 user.

The use of a negative forn (no) of the command disables filtering of received SNMPv3 packets.

Syntax

```
[no] ipv6 address <IPV6-ADDR>
```

Parameters

<IPV6-ADDR> – client IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-SNMP-USER

Example

```
esr(config-snmp-user)# ipv6 address AC:05:12:44::24
```

enable

This command enables SNMPv3 user.

The use of a negative form (no) of the command disables SNMPv3 user.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-SNMP-USER

Example

```
esr(config-snmp-user)# enable
```

oid-tree

This command sets OID and action applied to it (allow/deny). Longer OIDs have an advantage.

The OID is specified in numerical notation.

The use of a negative form (no) of the command removes the oid-tree entry.

Syntax

```
oid-tree <OID> <ACTION>
no oid-tree <OID>
```

Parameters

<OID> – OID, sets by string 255 characters;

<ACTION> – action applied to OID

- excluded – deny OID usage;
- included – allow OID usage.

Default value

Included.

Required privilege level

15

Command mode

CONFIG-SNMP-VIEW

Example

```
esr(config-snmp-view)# oid-tree 1.3.6.1.2.1.2.2 excluded
```

port

This command defines SNMP notifications collector port on the remote server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
port <PORT>
no port
```

Parameters

<PORT> – UDP port number, set in the range of [1..65535].

Default value

162

Required privilege level

15

Command mode

CONFIG-SNMP-HOST

Example

```
esr(config-snmp-host)# port 5555
```

privacy algorithm

This command defines encryption algorithm of transmitted data.

The use of a negative form (no) of the command disables the encryption.

Syntax

```
privacy algorithm <ALGORITHM>
no privacy algorithm
```

Parameters

<ALGORITHM> – encryption algorithm:

- aes128 – use AES-128 encryption algorithm;
- des – use DES encryption algorithm.

Required privilege level

15

Command mode

CONFIG-SNMP-USER

Example

```
esr(config-snmp-user)# privacy algorithm des
```

privacy key

This command sets a password for encryption of transmitted data.

The use of a negative form (no) of the command removes the password.

Syntax

```
privacy key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no privacy key
```

Parameters

<CLEAR-TEXT> – password, sets by string from 8 to 16 characters;

<ENCRYPTED-TEXT> – encrypted password from 8 bytes to 16 bytes (16 to 32 characters) in hexadecimal format (0xYYYY...) or (YYYY...).

Required privilege level

15

Command mode

CONFIG-SNMP-USER

Example

```
esr(config-snmp-user)# privacy key ascii-text 123456789
esr(config-snmp-user)# privacy key ascii-text encrypted CDE65039E5591FA3F1
```

rmon collection statistics

This command enables RMON statistics saving for physical interface.

The use of a negative form (no) of the command disables RMON statistics saving for physical interface.

Syntax

```
rmon collection statistics <INDEX> owner <OWNER>
no rmon collection statistics
```

Parameters

<INDEX> – specified interface RMON index;

<OWNER> – text field with [1..127] characters length that describes owner, that created this process.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example

```
esr(config-if)# rmon collection statistics 17 owner admin
```

snmp-server

This command enables the SNMP server in the global routing table as well as in all created VRFs.

The use of a negative form (no) of the command disables SNMP server.

Syntax

```
[no] snmp-server
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# snmp-server
```

snmp-server community

This command defines access community using the SNMP protocol.

The use of a negative form (no) of the command removes a community configurations.

Syntax

```
[no] snmp-server community <COMMUNITY> [ <TYPE> ] [ { <ADDR> | <IPV6-ADDR> } ] [client-list <OBJ-GROUP-NETWORK-NAME> ] [ <VERSION> ] [ view <VIEW-NAME> ] [ vrf <VRF> ]
```

Parameters

<COMMUNITY> – community for access via SNMP, set by a string [1..128] characters long;

<TYPE> – access level:

- ro – read only;
- rw – read/write.

<ADDR> – IP address of client that have access, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<IPV6-ADDR> – client IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

<OBJ-GROUP-NETWORK-NAME> – profile name of IP addresses, from which snmp requests are processing, set by the string of up to 31 characters.

<VERSION> – the snmp version supported by this community takes the values v1 or v2c.

<VIEW-NAME> – name of SNMP view profile, on which based access to OID.

<VRF> – VRF instance name, set by the string of up to 31 characters, for which access will be granted.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# snmp-server community public rw
```

snmp-server contact

This command sets SNMP variable value, that contains contact information (doesn't defined by default). For convenience, you can specify the person responsible for the equipment, such as his last name, in the parameters.

The use of a negative form (no) of the command removes SNMP variable value, that contains contact information.

Syntax

```
[no] snmp-server contact <CONTACT>
```

Parameters

<CONTACT> – contact information, sets by string with 255 characters length.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# snmp-server contact ivanov_ivan
```

snmp-server dscp

The command sets the DSCP code value for the use in IP headers of SNMP server outgoing packets.

The use of a negative form (no) of the command sets the default DSCP value.

Syntax

```
snmp-server dscp <DSCP>
```

```
no snmp-server dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Default value

61

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server dscp 40
```

snmp-server enable traps

This command allows sending all types of SNMP notifications.

The use of a negative form (no) of the command forbids sending all types of SNMP notifications.

Syntax

```
[no] snmp-server enable traps
```

Parameters

None.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps
```

snmp-server enable traps config

This command allows sending SNMP notifications about configuration operations.

The use of a negative form (no) of the command forbids sending SNMP notifications about configuration operations.

Syntax

```
[no] snmp-server enable traps config [ <ACT> ]
```

Parameters

<ACT> – configuration change fact traps:

- commit – configuration change appliance;
- confirm – configuration change confirmance.

Without specifying the <ACT> key – activates sending of all traps of this group.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps config commit
```

snmp-server enable traps entity

This command allows sending SNMP notifications about running-config operations.

The use of a negative form (no) of the command forbids sending SNMP notifications about configuration operations.

Syntax

```
[no] snmp-server enable traps entity [ <ENT> ]
```

Parameters

<ENT> – types of environment parameter filters:

- config-change – running-config operations information.

Without specifying the <ENT> key – activates sending of all traps of this group.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps entity
```

snmp-server enable traps entity-sensor

This command allows sending SNMP notifications about environment parameters changes.

The use of a negative form (no) of the command forbids sending SNMP notifications about configuration operations.

Syntax

```
[no] snmp-server enable traps entity-sensor [ <ENT> ]
```

Parameters

<ENT> – types of environment parameter filters:

- threshold – information on triggering threshold crossing.

Without specifying the <ENT> key – activates sending of all traps of this group.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps entity-sensor
```

snmp-server enable traps environment

This command allows sending SNMP notifications about environment parameters changes.

The use of a negative form (no) of the command forbids sending SNMP notifications about configuration operations.

Syntax

```
[no] snmp-server enable traps environment [ <ENV> ]
```

Parameters

<ENV> – types of environment parameter filters:

- pwrin – power supply failure;
- pwrin-insert – power supply installed;
- fan – fan failure;
- fan-speed-changed – fan speed changed;
- fan-speed-high – fan rotating speed exceeded the maximal threshold;
- memory-flash-low – NAND free space less than specified threshold;
- memory-flash-critical-low – NAND free space less than specified critical threshold;
- memory-ram-low – RAM free space less than specified maximal threshold;
- memory-ram-critical-low – RAM free space less than specified critical threshold;
- cpu-load – high CPU load;
- cpu-overheat-temp – CPU temperature exceeded specified maximal threshold;
- cpu-critical-temp – CPU temperature exceeded specified critical threshold;

- cpu-supercooling-temp – CPU temperature is lower than specified minimal threshold;
- switch-overheat-temp – switch temperature exceeded specified maximal threshold;
- switch-supercooling-temp – switch temperature is lower than specified minimal threshold;
- board-overheat-temp – board overheat;
- board-supercooling-temp – board supercooling;
- sfp-overheat-temp – SFP module overheat;
- sfp-supercooling-temp – SFP module supercooling.

Without specifying the <ENV> key – activates sending of all traps of this group.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps enviroment pwrin
```

snmp-server enable traps envmon

This command allows sending SNMP notifications about environment parameters changes.

The use of a negative form (no) of the command forbids sending SNMP notifications about configuration operations.

Syntax

```
[no] snmp-server enable traps envmon [ <ENV> ]
```

Parameters

<ENV> – types of environment parameter filters:

- fan – information on the operation of fan blocks;
- shutdown – information about disconnecting the router;
- supply – information about the operation of power supplies;
- temperature – information about the operation of temperature sensors.

Without specifying the <ENV> key – activates sending of all traps of this group.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps envmon fun
```

snmp-server enable traps files-operations

This command allows sending SNMP notifications about file operations.

The use of a negative form (no) of the command forbids sending SNMP notifications about configuration operations.

Syntax

```
[no] snmp-server enable traps files-operations [ <ACT> ]
```

Parameters

<ACT> – types of file operation parameter filters:

- successful;
- failed;
- canceled – canceled;

canceled – canceled.

Without specifying the <ACT> key – activates sending of all traps of this group.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps files-operations canceled
```

snmp-server enable traps flash

This command allows sending SNMP notifications about operations with external flash drives.

The use of a negative form (no) of the command forbids sending SNMP notifications about configuration operations.

Syntax

```
[no] snmp-server enable traps flash [ <ACT> ]
```

Parameters

<ACT> – types of file operation parameter filters:

- insertion – flash drive connection;
- removal – flash drive removal.

Without specifying the <ACT> key – activates sending of all traps of this group.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps flash removal
```

snmp-server enable traps interfaces

This command allows sending SNMP notifications about interface status changes.

The use of a negative form (no) of the command forbids sending SNMP notifications about configuration operations.

Syntax

```
[no] snmp-server enable traps interfaces [ <ACT> ]
```

Parameters

<ACT> – types of environment parameter filters:

- rx-utilization-high – incoming data stream exceeds threshold;
- tx-utilization-high – outgoing data stream exceeds threshold;
- number-high – excess number of IP interfaces;

Without specifying the <ACT> key – activates sending of all traps of this group.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps interfaces rx-utilization-high
```

snmp-server enable traps ports

This command allows sending SNMP notifications about errors on the interfaces of the switching chip.

The use of a negative form (no) of the command forbids sending SNMP notifications about configuration operations.

Syntax

```
[no] snmp-server enable traps ports [ <TYPE> ]
```

Parameters

<TYPE> – types of port status filters:

- port-counters-errors – errors on the switching chip interfaces.

Without specifying the <TYPE> key – activates sending of all traps of this group.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps ports
```

snmp-server enable traps screens

This command allows broadcast of SNMP notifications about protection from a certain type of DoS attacks.

The use of a negative form (no) of the command forbids sending SNMP notifications about configuration operations.

Syntax

```
[no] snmp-server enable traps screens [ <SCREEN> ]
```

Parameters

<SCREEN> – types of DoS attacks protection filters:

- dest-limit – limiting the number of simultaneous sessions based on the destination address;
- source-limit – limiting the number of simultaneous sessions based on the source address;

- icmp-threshold – protection from ICMP flood attacks;
- udp-threshold – protection from UDP flood attacks;
- syn-flood – protection from SYN flood attacks;
- land – protection from land attacks;
- winnuke – protection from winnuke attacks;
- icmp-frag – fragmented ICMP packets blocking;
- syn-flag – fragmented TCP packets blocking, with SYN flag;
- unknown-proto – blocking of packets, with the protocol ID contained in IP header equal to 137 and more;
- ip-frag – fragmented packets blocking;
- port-scan – protection from port scan attacks;
- ip-sweep – protection from IP-sweep attacks;
- syn-fin – blocking of TCP packets, with the SYN and FIN flags set;
- fin-no-ack – blocking of TCP packets with the FIN flag set and the ACK flag not set;
- no-flag – blocking of TCP packets with the null flag field;
- spoofing – protection from IP spoofing attacks;
- reserved – blocking of all ICMP packets of types 2 and 7 (reserved);
- quench – blocking of all ICMP packets of type 4 (source quench);
- echo-request – blocking of all ICMP packets of type 8 (echo-request);
- time-exceeded – blocking of all ICMP packets of type 11 (time exceeded);
- unreachable – blocking of all ICMP packets of type 3 (destination-unreachable);
- icmp-large – blocking ICMP packets with large size;
- tcp-all-flags – blocking tcp packets with flags;
- udp-frag – blocking udp packets with flags.

Without specifying the <LINK> key – activates sending of all traps of this group.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps screens reserved
```

snmp-server enable traps snmp

This command allows sending SNMP notifications about environment parameters changes.

The use of a negative form (no) of the command forbids sending SNMP notifications about configuration operations.

Syntax

```
[no] snmp-server enable traps snmp [ <ACT> ]
```

Parameters

<ACT> – types of environment parameter filters:

- authentication – notifications about snmp requests to the router with the wrong community or snmpv3 password;
- coldstart – notifications about restarting the snmp server on the router;
- linkdown – information about link status change to down;
- linkup – information about link status change to up;

Without specifying the <ACT> key – activates sending of all traps of this group.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps snmp linkup
```

snmp-server enable traps syslog

This command allows sending SNMP notifications with syslog messages.

The use of a negative form (no) of the command forbids sending SNMP notifications about configuration operations.

Syntax

```
[no] snmp-server enable traps syslog
```

Parameters

None.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps syslog
```

snmp-server enable traps wifi

This command allows SNMP notifications to be sent with messages about the operation of the Wi-Fi controller.

The use of a negative form (no) of the command forbids sending SNMP notifications about configuration operations.

Syntax

```
[no] snmp-server enable traps wifi [ <NAME> ]
```

Parameters

<NAME> – types of softgre tunnels traps:

- wifi-tunnels-number-in-bridge-high – enable traps on exceeding the number of sub-gre tunnels included in the bridge
- wifi-tunnels-operation – enable traps on the result of snmp operations with softgre tunnels.

Without specifying the <NAME> key – activates sending of all traps of this group.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server enable traps syslog
```

snmp-server host

This command enables the transmission of SNMP notifications to the specified IP address and switches to the SNMP notifications configuration mode.

The use of a negative form (no) of the command disables the transmission of notifications to the specified SNMP notification collector.

Syntax

```
[no] snmp-server host { <ADDR> | <IPV6-ADDR> } [vrf <VRF>]
```

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<VRF> – VRF instance name, set by the string of up to 31 characters, which contains SNMP notification collector.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# snmp host 192.168.2.2
```

snmp-server location

This command sets SNMP variable value, that contains information about equipment location (doesn't defined by default). For convenience, you can specify the city, street, district, room number, etc. in the parameters.

The use of a negative form (no) of the command removes the value of the variable containing the equipment location information.

Syntax

```
[no] snmp-server location <LOCATION>
```

Parameters

<LOCATION> – information about equipment location, set by the string up to 255 characters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# snmp-server location duglasa_adamsa_42
```

snmp-server system-shutdown

This command allows the router to be rebooted using snmp messages.

The use of a negative form (no) of the command denies the router to be rebooted using snmp messages.

Syntax

```
[no] snmp-server system-shutdown
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# snmp-server system-shutdown
```

snmp-server trap link

This command sets the send mode of SNMP-trap.

The use of a negative form (no) of the command sets the default mode.

Syntax

```
snmp-server trap link <MODE>
no snmp-server host
```

Parameters

<MODE> – SNMP-trap transmission mode. Takes the following values:

- ietf;
- cisco.

Default value

ietf

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# snmp-server trap link cisco
```

snmp-server user

This command creates SNMPv3 user.

The use of a negative form (no) of the command removes SNMPv3 user.

Syntax

```
[no] snmp-server user <NAME>
```

Parameters

<NAME> – user name, set by the string from 1 to 128 characters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# snmp-server user admin
esr(config-snmp-user)#
```

snmp-server view

This command creates a snmp view profile that allows you to allow or deny access to certain OIDs for the community (SNMPv2) and user (SNMPv3).

The use of a negative form (no) of the command removes snmp view profile.

Syntax

```
[no] snmp-server view <VIEW-NAME>
```

Parameters

<VIEW-NAME> – SNMP view profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# snmp-server view user_access
esr(config-snmp-view)#
```

source-address

This command defines the IP address to send notifications to the remote server.

The use of a negative form (no) of the command sets the default value.

Syntax

`source-address { <ADDR> | <IPV6-ADDR> }`

`no source-address`

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Default value

IPv4/IPv6 – The address of the interface closest to the remote SNMP server.

Required privilege level

15

Command mode

CONFIG-SNMP-HOST

Example

```
esr(config-snmp-host)# source-address 192.168.22.17
```

source-interface

This command defines the interface or tunnel of the router whose IPv4/IPv6 address will be used to send notifications to the remote server.

The use of a negative form (no) of the command removes a specified interface or tunnel.

Syntax

```
source-interface { <IF> | <TUN> }
```

```
no source-interface
```

Parameters

<IF> – an interface's name, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

Required privilege level

15

Command mode

CONFIG-SNMP-HOST

Example

```
esr(config-snmp-host)# source-interface gigabitethernet 1/0/1
```

view

This command sets a snmp view profile that allows you to allow or deny access to certain OIDs for SNMPv3 user.

The use of a negative form (no) of the command removes snmp view profile.

Syntax

```
[no] view <VIEW-NAME>
```

Parameters

<VIEW-NAME> – name of SNMP view profile, on which based access to OID, set by the string up to 31 characters.

Required privilege level

15

Command mode

CONFIG-SNMP-USER

Example

```
esr(config-snmp-user)# view user_view
```

SYSLOG management

- logging aaa configuration
- logging acl configuration
- logging firewall configuration
- logging login on-failure
- logging nat
- logging service start-stop
- logging syslog configuration
- logging userinfo
- ntp logging
- show syslog
- show syslog configuration
- syslog cli-commands
- syslog console
- syslog file
- syslog file-size
- syslog host
- syslog max-files
- syslog monitor
- syslog reload debugging
- syslog sequence-numbers
- syslog snmp
- syslog timestamp msec
- syslog sip-level

logging aaa configuration

This command enables writing to the local syslog server of messages about aaa partition configuration changes.

The use of a negative form (no) of the command disables syslog aaa partition changes logging.

Syntax

```
[no] logging aaa configuration
```

Parameters

None.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# logging aaa configuration
```

logging acl configuration

This command enables writing to the local syslog server of messages about ACL configuration changes.

The use of a negative form (no) of the command disables ACL configuration changes logging.

Syntax

```
[no] logging acl configuration
```

Parameters

None.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# logging acl configuration
```

logging firewall configuration

This command enables writing to the local syslog server of messages about firewall configuration changes.

The use of a negative form (no) of the command disables firewall configuration changes logging.

Syntax

```
[no] logging firewall configuration
```

Parameters

None.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# logging firewall configuration
```

logging login on-failure

This command enables writing to the local syslog server of messages about unsuccessful attempts to connect to the CLI.

The use of a negative form (no) of the command disables unsuccessful connection attempts logging.

Syntax

`[no] logging login on-failure`

Parameters

None.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# logging login on-failure
```

logging nat

This command enables writing to the local syslog server of messages about NTP service operation.

The use of a negative form (no) of the command disables NAT service logging.

Syntax

```
[no] logging nat [<NAT-TYPE>]
```

Parameters

<NAT-TYPE> – type of NAT service to be logged:

- destination;
- proxy;
- source;

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# logging nat source
```

logging service start-stop

This command enables writing to the local syslog server of messages about starting and stopping used services.

The use of a negative form (no) of the command disables start and stop services used logging.

Syntax

```
[no] logging service start-stop
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# logging service start-stop
```

logging syslog configuration

This command enables writing to the local syslog server of messages about syslog server configuration changes.

The use of a negative form (no) of the command disables syslog server configuration changes logging.

Syntax

```
[no] logging syslog configuration
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# logging syslog configuration
```

logging userinfo

This command enables writing to the local syslog server of messages about user-profile changes.

The use of a negative form (no) of the command disables user-profile changes logging.

Syntax

```
[no] logging userinfo
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# logging userinfo
```

ntp logging

This command enables writing to the local syslog server of messages about operations with NTP peers.

The use of a negative form (no) of the command disables NTP peers operations logging.

Syntax

```
[no] ntp logging
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ntp logging
```

show syslog

This command displays the current configuration information of the syslog, a list of created log files, as well as to view the log files with the ability to filter using regular expressions.

Syntax

```
show syslog <FILE> [ from-date <YEAR> <MONTH> <DAY> ] [ from-time <TIME> ] [ to-date
<YEAR> <MONTH> <DAY> ] [ to-time <TIME> ] [ from-end ]
```

Parameters

<FILE> – file name, set by the string of up to 31 characters;

from-date – to display information from the specified date;

from-time – to display information from the specified time;

to-date – to display information before the specified date;

to-time – to display information before the specified time;

<YEAR> – year, takes values of [2001..2037].

<MONTH> – month, takes the following values [January/February/March/April/May/June/July/August/September/October/November/December];

<DAY> – day of the month, takes values of [1..31];

<TIME> – system timer, defined as HH MM SS, where:

- HH – hours, takes the value of [0..23];
- MM – minutes, takes the value of [0 ..59];
- SS – seconds, takes the value of [0..59];

from-end – viewing the contents of the file from the end, as the last entries are placed at the end of the file;

Required privilege level

15

Command mode

ROOT

Example

```
esr# show syslog
   Log files
   ~~~~~~~~~
##     Name                   Size in bytes      Date of last modification
----   --------------------   ----------------   -------------------------
1      debug                  371681             Thu Jan  1 16:17:04 1970
2      debug.1                524222             Thu Jan  1 01:48:13 1970
3      esr                    97259              Thu Jan  1 16:17:01 1970
----   --------------------   ----------------   -------------------------
Total files: 4
esr# show syslog configuration
SYSLOG
File size: 512 (kiB)
Number of logs: 3
Console: info
   Files:
   ~~~~~~
ID   Name                               Severity
--   --------------------------------   ----------
0    esr                                info
```

show syslog configuration

This command allows to view current syslog log configuration information.

Syntax

```
show syslog configuration
```

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

ROOT

Example

```
esr# show syslog configuration

SYSLOG

File size: 500 (kiB)
Number of logs: 1
Console: info
Monitor: info
```

syslog cli-commands

This command enables the process of logging user input commands to a local syslog server.

The use of a negative form (no) of the command disables command logging.

Syntax

```
[no] syslog cli-commands
```

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# syslog cli-commands
```

syslog console

This command sets the levels of syslog messages that will be displayed in the console. Displays messages that have a severity level, specified in a command, or higher.

The use of a negative form (no) of the command sets the default level of displayed messages.

Syntax

```
syslog console <SEVERITY>
```

```
no syslog console
```

Parameters

<SEVERITY> – message importance level, takes values (in order of decreasing importance):

- emerg – critical error has occurred in the system, the system is not operational;
- alert – alarms, immediate intervention by staff;
- crit – critical system status, event reporting;
- error – error messages;
- warning – warnings, non-emergency messages;
- notice – messages about important system events;
- info – system information messages;
- debug – debugging messages provide the user with information to correctly configure the system;
- none – disables the output of syslog messages to the console.

Default value

info

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# syslog console info
```

syslog file

This command enables the saving of syslog messages of a specified level of importance to the specified log file. Saves messages that have a severity level, specified in a command, or higher.

The use of a negative form (no) of the command disables the saving of syslog messages in specified file.

Syntax

```
syslog file { flash:syslog/<NAME> | tmpsys:syslog/<NAME> } <SEVERITY>
no syslog file { flash:syslog/<NAME> | tmpsys:syslog/<NAME> | all}
```

Parameters

flash – file is located in the non-volatile memory of the device;

tmpsys – file is located in the volatile memory of the device;

<NAME> – name of the file to which messages of a given level will be recorded, specified by the string up to 31 characters. The use of a negative form (no) of the command with 'all' parameter removes all configured syslog files.

<SEVERITY> – importance level of the message, possible values are given in section syslog console.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# syslog file esr info
```

syslog file-size

This command sets the maximal log file size. If the specified size is exceeded, files will be rotated automatically.

The use of a negative form (no) of the command sets the default value of log file size.

Syntax

```
syslog file-size <SIZE>
no syslog file-size
```

Parameters

<SIZE> – file size, takes the value [10..10000000] KB.

Default value

500 KB

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# syslog file-size 10000
```

syslog host

This command enables the sending of syslog messages of a specified level of importance to a remote syslog server. Sends messages that have a severity level, specified in a command, or higher.

The use of a negative form (no) of the command disables the sending of syslog messages to a remote syslog server.

Syntax

```
syslog host <HOSTNAME> { <ADDR> | <IPV6-ADDR> } [ <SEVERITY> ] [ <TRANSPORT> ] [ <PORT> ]
[ vrf <VRF> ] [ source-address { <SRC-ADDR> | <IPV6-SRC-ADDR> } ]
```

```
no syslog host { <HOSTNAME> | all }
```

Parameters

<HOSTNAME> – syslog server name, set by the string of up to 31 characters. Used only to identify the server during configuration. The value 'all' is used in the **no syslog host all** command to delete all syslog servers;

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IPV6-ADDR> – IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<SEVERITY> – importance level of the message, optional parameter, possible values are given in section syslog console;

<TRANSPORT> – data transfer protocol, optional parameter, takes values:

- TCP – data transmission is carried out by TCP;
- UDP – data transmission is carried out by UDP;

<PORT> – number of TCP/UDP port, optional parameter, takes values of [1..65535], default value is 514;

<VRF> – VRF instance name, set by the string of up to 31 characters, which contains remote syslog server.

<SRC-ADDR> – IP address of the router that will be used as the source IP address in sent syslog packets.

<IPV6-SRC-ADDR> – IPv6 address of the router that will be used as the source IPv6 address in sent syslog packets;

all – key to delete all configured syslog-host.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# syslog host eltex 192.168.2.2
```

syslog max-files

This command sets the maximum number of files saved during rotation.

The use of a negative form (no) of the command sets the default value of saved files amount.

Syntax

```
syslog max-files <NUM>
no syslog max-files
```

Parameters

<NUM> – maximal numberf of files , takes values [1..1000].

Default value

15

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# syslog max-files 100
```

syslog monitor

This command sets the level of syslog messages that will be displayed during remote connections (Telnet, SSH). Displays messages that have a severity level, specified in a command, or higher.

The use of a negative form (no) of the command sets the default level of displayed messages.

Syntax

```
syslog monitor <SEVERITY>
no syslog monitor
```

Parameters

<SEVERITY> – importance level of the message, possible values are given in section syslog console.

Default value

info

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# syslog monitor info
```

syslog reload debugging

This command enables debugging mode during the reboot process for the local syslog server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] syslog reload debugging
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# syslog reload debugging
```

syslog sequence-numbers

This command enables numbering of the records in the local syslog-server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] syslog sequence-numbers
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# syslog sequence-numbers
```

syslog snmp

This command sets the level of syslog messages that will be sent to the snmp server in the form of snmp-trap. The messages that have a severity level, specified in a command, or higher are sent.

The use of a negative form (no) of the command sets the default level of displayed messages.

Syntax

```
syslog snmp <SEVERITY>
no syslog snmp
```

Parameters

<SEVERITY> – importance level of the message, possible values are given in section syslog console.

Default value

info

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# syslog snmp info
```

syslog timestamp msec

⚠ In the current firmware version this functionality is supported by ESR-100/200/1000/1200/1500/1511/1700/3100 routers only

This command enables adding the milliseconds to time to the records in the local syslog-server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] syslog timestamp msec
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# syslog timestamp msec
```

syslog sip-level

This command sets the level of sip event logging.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] syslog sip-level <LEVEL>
```

Parameters

<LEVEL> – the level of sip messages that will go into the syslog. Takes values in the range of [0..9].

Default value

0

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# syslog sip-level 8
```

SSH, Telnet acces configuration

- crypto key generate
- ip ftp client password
- ip ftp client username
- ip sftp client password
- ip sftp client username
- ip ssh authentication algorithm disable
- ip ssh authentication retries
- ip ssh authentication timeout
- ip ssh client password
- ip ssh client source-ip
- ip ssh client username
- ip ssh dscp
- ip ssh encryption algorithm disable
- ip ssh key-exchange algorithm disable
- ip ssh key-exchange time
- ip ssh key-exchange volume
- ip ssh port
- ip ssh server
- ip telnet dscp
- ip telnet port
- ip telnet server
- show crypto key mypubkey

crypto key generate

This command generates a pair of cryptographic keya to establish SSH connection.

Syntax

```
crypto key generate [ dsa | escda <ESCDA> | ed25519 <ED25519> | rsa <RSA> | rsa1 <RSA1> ]
```

Parameters

dsa – DSA algorithm;

ecdsa – ECDSA algorithm;

- <ECDSA> – key size, takes the value 256, 384 or 521;
- Without specification, key size 521 is used.

ed25519 – ED25519 algorithm;

- <ED25519> – key size, may take values [256..2048];
- Without specification, key size 2048 is used.

rsa – RSA algorithm with specifying the key length;

- <RSA> – key size, may take values [1024..2048];
- Without specification, key size 2048 is used.

rsa1 – RSA1 algorithm.

- <RSA> – key size, may take values [1024..2048];
- Without specification, key size 2048 is used.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# crypto key generate ecdsa
```

ip ftp client password

This command defines the default password for FTP copy operations.

The use of a negative form (no) of the command removes the password.

Syntax

```
ip ftp client password { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
[no] ftp client password
```

Parameters

<CLEAR-TEXT> – password, set by the string of 1 to 16 characters, takes the value of [0-9a-fA-F];

<ENCRYPTED-TEXT> – encrypted password, set by the string of [2..32] characters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip ftp client password test
```

ip ftp client username

This command defines the default user name for FTP copy operations.

The use of a negative form (no) of the command removes a user name.

Syntax

```
ip ftp client username <NAME>
```

```
no ftp client username
```

Parameters

<NAME> – user name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip ftp client username test
```

ip sftp client password

This command sets the password values for an SFTP client.

The use of a negative form (no) of the command removes the password values for an SFTP client.

Syntax

```
ip sftp client password { <TEXT> | encrypted < ENCRYPTED-TEXT > }
no ip sftp client password
```

Parameters

<TEXT> – string [1..16] ASCII characters;

<ENCRYPTED-TEXT> – encrypted password, [8..16] bytes size, set by the string of [16..32] characters.

Default value

Username is not specified.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip sftp client password 123456789
```

ip sftp client username

This command sets the user name values for an SFTP client.

The use of a negative form (no) of the command removes the user name values for an SFTP client.

Syntax

```
ip sftp client username <USERNAME>
no ip sftp client username
```

Parameters

<USERNAME> – user name, set by the string of up to 31 characters.

Default value

Username is not specified.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip sftp client username esruser
```

ip ssh authentication algorithm disable

This command prohibits the use of a specific authentication algorithm for SSH server.

The use of a negative form (no) of the command allows the use of a specific authentication algorithm for the SSH server.

Syntax

```
[no] ip ssh authentication algorithm <ALGORITHM> disable
```

Parameters

<ALGORITHM> – authentication algorithm, takes values of: [md5, md5-96, sha1, sha1-96, sha2-256, sha2-512, ripemd160].

Required privilege level

15

Default value

Allow all authentication algorithms.

Command mode

CONFIG

Example

```
esr(config)# no ip ssh authentication algorithm md5 disable
```

ip ssh authentication retries

This command sets the number of authentication attempts for SSH server.

The use of a negative form (no) of the command sets the default number of authentication attempts for SSH server.

Syntax

```
ip ssh authentication retries <NUM>
no ip ssh authentication retries
```

Parameters

<NUM> – number of authentication attempts for SSH server [1..10].

Required privilege level

10

Default value

6

Command mode

CONFIG

Example

```
esr(config)# ip ssh authentication retries 5
```

ip ssh authentication timeout

This command sets authentication timeout period for SSH server.

The use of a negative form (no) of the command sets the default authentication timeout period for SSH server.

Syntax

```
ip ssh authentication timeout <SEC>
no ip ssh authentication timeout
```

Parameters

<SEC> – time interval in seconds, takes values of [30..360].

Required privilege level

10

Default value

120

Command mode

CONFIG

Example

```
esr(config)# ip ssh authentication timeout 60
```

ip ssh client password

This command defines the default password for SCP copy operations.

The use of a negative form (no) of the command removes the password.

Syntax

```
ip ssh client password { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no ssh client password
```

Parameters

<CLEAR-TEXT> – password, set by the string of 1 to 16 characters, takes the value of [0-9a-fA-F];

<ENCRYPTED-TEXT> – encrypted password, set by the string of [2..32] characters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip ssh client password test132
```

ip ssh client source-ip

This command defines the ip-address of the router from which ssh sessions will be established on other devices.

The use of a negative form (no) of the command removes a user name.

Syntax

```
ip ssh client source-ip <ADDR>
no ssh client source-ip
```

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. The specified IP address must be assigned on any interface/tunnel of the router.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip ssh client source-ipt 192.168.22.78
```

ip ssh client username

This command defines the default user name for SCP copy operations.

The use of a negative form (no) of the command removes a user name.

Syntax

```
ip ssh client username <NAME>
no ssh client username
```

Parameters

<NAME> – user name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip ssh client username tester
```

ip ssh dscp

The command sets the DSCP code value for the use in IP headers of SSH server outgoing packets.

The use of a negative form (no) of the command sets the default DSCP value.

Syntax

```
ip ssh dscp <DSCP>
no ip ssh dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Default value

32

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip ssh dscp 40
```

ip ssh encryption algorithm disable

This command prohibits the use of a specific encryption algorithm for SSH server.

The use of a negative form (no) of the command allows the use of a specific encryption algorithm for the SSH server.

Syntax

```
[no] ip ssh encryption algorithm <ALGORITHM> disable
```

Parameters

<ALGORITHM> – encryption algorithm identifier, takes values [aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, arcfour, arcfour128, arcfour256, blowfish, cast128, 3des].

Required privilege level

15

Default value

All algorithms are allowed.

Command mode

CONFIG

Example

```
esr(config)# ip ssh encryption algorithm aes128 disable
```

ip ssh key-exchange algorithm disable

This command prohibits the use of a specific key exchange algorithm for SSH server.

The use of a negative form (no) of the command allows the use of a specific key exchange algorithm for the SSH server.

Syntax

```
[no] ip ssh key-exchange algorithm <ALGORITHM> disable
```

Parameters

<ALGORITHM> – key exchange protocol identifier, takes values [dh-group1-sha1, dh-group14-sha1, dh-group-exchange-sha1, dh-group-exchange-sha256, ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521].

Required privilege level

15

Default value

All algorithms are allowed.

Command mode

CONFIG

Example

```
esr(config)# ip ssh key-exchange algorithm dh-group-exchange-sha1 disable
```

ip ssh key-exchange time

This command sets authentication keys changing period for SSH server.

The use of a negative form (no) of the command sets the default authentication keys changing period for SSH server.

Syntax

```
ip ssh key-exchange time <SEC>
no ip ssh key-exchange time
```

Parameters

<SEC> – time interval in hours, takes values of [1..72].

Required privilege level

15

Default value

1

Command mode

CONFIG

Example

```
esr(config)# ip ssh key-exchange time 24
```

ip ssh key-exchange volume

This command sets the amount of data, after passing which, the authentication keys for the SSH server will be updated.

The use of a negative form (no) of the command sets the amount of data, after passing which, the authentication keys for the default SSH server will be updated.

Syntax

```
ip ssh key-exchange volume <DATA>
no ip ssh key-exchange volume
```

Parameters

<DATA> – data size in MB, takes values [1..4096].

Required privilege level

15

Default value

1000

Command mode

CONFIG

Example

```
esr(config)# ip ssh key-exchange volume 512
```

ip ssh port

This command defines the SSH server port on the router.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip ssh port <PORT>
no ip ssh port
```

Parameters

<PORT> – port number, set in the range of [1..65535].

Default value

22

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip ssh port 3001
```

ip ssh server

This command enables the SSH server on the router.

The use of a negative form (no) of the command disables SSH server.

Syntax

```
[no] ip ssh server [ vrf <VRF>]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the SSH server will operate.

Default value

SSH server is disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# no ip ssh server
```

ip telnet dscp

The command sets the DSCP code value for the use in IP headers of Telnet server outgoing packets.

The use of a negative form (no) of the command sets the default DSCP value.

Syntax

```
ip telnet dscp <DSCP>
no ip telnet dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Default value

32

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip telnet dscp 40
```

ip telnet port

This command defines the Telnet server port on the router.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip telnet port <PORT>
no ip telnet port
```

Parameters

<PORT> – port number, takes values of [1..65535].

Default value

23

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# ip telnet port 2001
```

ip telnet server

This command enables the Telnet server on the router.

The use of a negative form (no) of the command disables Telnet server.

Syntax

```
[no] ip telnet server [vrf <VRF>]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the Telnet server will operate.

Default value

Telnet server is disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# no ip telnet server
```

show crypto key mypubkey

The command displays the device's public keys used to establish an SSH connection.

Syntax

```
show crypto key mypubkey <OPTIONS>
```

Parameters

<OPTIONS> – algorithm for generating a new cryptographic key:

- dsa – DSA algorithm;
- ecdsa – ECDSA algorithm;
- ed25519 – ED25519 algorithm;

- rsa – RSA algorithm;
- rsa1 – RSA1 algorithm.

Required privilege level

15

Command mode

ROOT

Example

```
esr# show crypto key mypubkey rsa
Key data
------------------------------------------------------------
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDz750sWCQrnNufg1yhuksTFYCYdEfg
JZ9tWUvcssAZhCJWMewprXBuZMABzFmfBg157pgapxn2qJXJ8ESMV7X7gPfy
xQQah6l376z3SFcpKvwudNgwHiS5HCYPRQWx2Xdaz/nJtYr5NpYgLPba68NC
iXcqEp7EPR5GojDVxpuDuk0hPFcihzmt5Yx8ZptJRzRtsuDQYlowv0Qa24kd
OlQ90/1qKfbAhB6XI60l+dK5VEj7giBESarcRn69/e/YVbdGBdTE93QWFPKI
bm63imfbxRwWtcwsFdIHi8Blv9ZqDqqF/IO3TkIKa31hV9GnsawlAXi/IdyY
bYPboHRdcTlH/ root@esr-1000
```

LLDP configuration

- lldp enable
- lldp hold-multiplier
- lldp management-address
- lldp med fast-start enable
- lldp med fast-start tx-interval
- lldp receive
- lldp system-description
- lldp system-name
- lldp timer
- lldp transmit
- show lldp neighbors
- show lldp statistics

lldp enable

This command enables LLDP protocol support on the router.

The use of a negative form (no) of the command disables LLDP protocol support.

Syntax

```
[no] lldp enable
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# lldp enable
```

lldp hold-multiplier

This command sets the number of lldp-timer periods during which the router stores information received via the LLDP protocol.

The use of a negative form (no) of the command sets the specified parameter to the default value.

Syntax

```
lldp hold-multiplier <SEC>
no lldp hold-multiplier
```

Parameters

<SEC> – time interval in seconds, takes values of [1..10].

Default value

4

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# lldp hold-multiplier 5
```

lldp management-address

This command sets the IP address that will be sent to the LLDP TLV as the management-address.

The use of a negative form (no) of the command sets the default value for the LLDP TLV management-address field.

Syntax

```
lldp management-address <ADDR>
no lldp management-address
```

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. The specified IP address must be assigned on any interface/tunnel of the router.

Default value

One of the existing.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# lldp management-address 192.168.54.42
```

lldp med fast-start enable

This command enables the sending of LLDP-MED messages on interfaces with a configured network policy (see section lldp network-policy).

The use of a negative form (no) of the command disables LLDP-MED broadcasting of messages on interfaces with a configured network policy.

Syntax

```
[no] lldp med fast-start enable
```

Parameters

None.

Default value

LLDP-MED messaging is disabled.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# lldp med fast-start enable
```

lldp med fast-start tx-interval

This command configures the period of sending of LLDP-MED messages on interfaces with a configured network policy (see section lldp network-policy).

The use of a negative form (no) sets the default value for the distribution period of LLDP-MED messages on interfaces with a configured network policy.

Syntax

```
lldp med fast-start tx-interval <SEC>
[no] lldp med fast-start tx-interval
```

Parameters

<SEC> – time interval, takes values of [1..32768] seconds.

Default value

1 second

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# lldp med fast-start tx-interval 20
```

lldp receive

This command enables LLDPDU processing on the physical interface.

The use of a negative form (no) of the command disables LLDPDU processing on the physical interface.

Syntax

```
[no] lldp receive
```

Parameters

The command does not contain parameters.

Default value

LLDP packets receiving disabled.

Required privilege level

15

Command mode

CONFIG-IF-GI

CONFIG-IF-TE

Example

```
esr(config-if-gi)# lldp receive
```

lldp system-description

This command sets the field that will be sent to the LLDP TLV as the system-description.

The use of a negative form (no) of the command sets the default value for the LLDP TLV system-description field.

Syntax

```
lldp system-description <DESCRIPTION>
no lldp system-description
```

Parameters

<DESCRIPTION> – system description identifier, set by the string of up to 255 characters.

Default value

Contains information about the model and firmware version of the router.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# lldp system-description esr-1000-1.3.0
```

lldp system-name

This command sets the field that will be sent to the LLDP TLV as the system-name.

The use of a negative form (no) of the command.

Syntax

```
lldp system-name <NAME>
no lldp system-name
```

Parameters

<NAME> – system name identifier, set by the string of up to 255 characters.

Default value

Same as hostname.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# lldp system-name esr-100-branch-12
```

lldp timer

This command sets the LLDPDU send period.

The use of a negative form (no) of the command sets the default LLDPDU send period value.

Syntax

```
lldp timer <SEC>
no lldp timer
```

Parameters

<SEC> – time interval in seconds, takes values of [1..32768].

Default value

30

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# lldp timer 3
```

lldp transmit

This command enables LLDPDU sending on the physical interface.

The use of a negative form (no) of the command disables LLDPDU sending on the physical interface.

Syntax

```
[no] lldp transmit
```

Parameters

The command does not contain parameters.

Default value

LLDP packets sending disabled.

Required privilege level

15

Command mode

CONFI-IF-GI

CONFI -IF-TE

Example

```
esr(config-if-gi)# lldp transmit
```

show lldp neighbors

This command displays information about the connected devices from which information is received via the LLDP protocol.

Syntax

```
show lldp neighbors <IF>
```

Parameters

<IF> – name of physical interface or a group of physical interfaces is specified in the form described in Section Types and naming order of router interfaces.

It is permissible to use only physical interfaces (gigabitethernet or tengigabitethernet). Without an interface, information about all devices detected by the LLDP is displayed.

Required privilege level

10

Command mode

ROOT

Example

```
esr# show lldp neighbors
LLDP Neighbor Information:

Local Information:
    Index:              0
    Local Interface:    gi1/0/4

Neighbour Information:
    Chassis type:       mac
    Chassis ID:         a8:f9:4b:aa:8c:90
    Management ip:      192.168.1.5
    Management ip:      fe80::2052:e5ff:fe36:226f
    Port type:          local
    Port ID:            gi1/0/8
    Port description:   esr200-lldp-test
    Time to live:       120
    System name:        esr-200-test
    System Description: Eltex Router ESR-200 1.3.0 build 79 (date 14/08/2017 time 10:19:13)

System capabilities:
    Bridge:             false
    Router:             true
    Station:            true
    Wlan:               false
```

show lldp statistics

This command displays the statistics of the LLDP protocol operation on the interfaces on which the LLDP protocol is enabled.

Syntax

```
show lldp statistics <IF>
```

Parameters

<IF> – name of an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces; It is permissible to use only physical interfaces (gigabitethernet or tengigabitethernet). Without an interface, information about all devices detected by the LLDP is displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show lldp statistics
Interface  Transmitted  Received  Discarded  Unrecognized  Ageout  Inserted   Deleted
---------  -----------  --------  ---------  -----------   ------  --------   -------
 gi1/0/1        1          0          0           0           0        0         0
```

Mirroring configuration

- port monitor
- port monitor interface
- port monitor mode
- port monitor remote
- port monitor remote vlan
- show interfaces switch-port monitor

port monitor

This command enables traffic mirroring mode.

The use of a negative form (no) of the command disables the traffic mirroring mode.

Syntax

```
[no] port monitor
```

Parameters

None.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example

```
esr(config-if-gi)# port monitor interface gigabitethernet 1/0/5
```

port monitor interface

This command defines controlling ports.

The use of a negative form (no) of the command removes controlling port.

Syntax

```
port monitor interface <IF> <DIRECTION>
```

```
no port monitor
```

Parameters

<IF> – physical interface or a group of physical interfaces is specified in the form described in Section Types and naming order of router interfaces;

<DIRECTION> – traffic direction:

- tx – only outgoing traffic mirroring;
- rx – only incoming traffic mirroring.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example

```
esr(config-if-gi)# port monitor interface gigabitethernet 1/0/5
```

port monitor mode

This command defines the mode of the port transmitting mirrored traffic.

The use of a negative form (no) of the command sets the default value.

Syntax

```
port monitor mode <MODE>
```

```
no port monitor mode
```

Parameters

<MODE> – mode:

- network – combined data transfer and mirroring;
- monitor-only – mirroring only.

Default value

network

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# port monitor mode monitor-only
```

port monitor remote

> ⚠ In the current firmware version this functionality is supported only by ESR-1000/1200/1500/1511/1700 routers

This command enables remote mirroring (RSPAN).

The use of a negative form (no) of the command disables remote mirroring (RSPAN).

Syntax

`[no] port monitor remote`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

Example

```
esr(config-if-gi)# port monitor remote
```

port monitor remote vlan

> ⚠ In the current firmware version this functionality is supported only by ESR-1000/1200/1500/1511/1700 routers.

This command defines the VLAN over which the mirrored traffic will be transmitted.

The use of a negative form (no) of the command removes a specified VLAN.

Syntax

```
port monitor remote vlan <VID> <DIRECTION>
no port monitor remote vlan <DIRECTION>
```

Parameters

<VID> – VLAN ID, set in the range of [2..4094];

<DIRECTION> – traffic direction:

- tx – mirroring only outgoing traffic to the specified VLAN;
- rx – mirroring only incoming traffic to the specified VLAN.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# port monitor remote vlan 10
```

show interfaces switch-port monitor

The commands displays mirroring settings.

Syntax

```
show interfaces switch-port monitor
```

Parameters

None.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show interfaces switch-port monitor
Port monitor mode:          network
RSPAN configuration RX:     VLAN 222
RSPAN configuration TX:     VLAN 222
Source Port    Destination Port   Type     RSPAN
------------   ----------------   ------   --------
gi1/0/7        gi1/0/6            RX,TX    Enabled
```

Zabbix-agent and zabbix-proxy configuration

- active-server
- config-retrieve
- database
- enable
- hostname
- port
- remote-commands
- server
- server
- show zabbix-agent configuration
- show zabbix-proxy configuration
- source-address
- timeout
- zabbix-agent
- zabbix-proxy

active-server

This command sets the address and port of the Zabbix server for active checks.

The use of a negative form (no) of the command removes server for active checks.

Syntax

```
active-server <ADDR> [ port <PORT> ]
```

```
no active-server
```

Parameters

<ADDR> – server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<PORT> – port number, set in the range of [1..65535].

Default value

None

Required privilege level

10

Command mode

CONFIG-ZABBIX-AGENT

Example

```
esr(config-zabbix-agent)# active-server 192.168.16.54
```

config-retrieve

This command is used to set the interval for updating configuration data from the server.

The use of a negative form (no) of the command restores the default value.

Syntax

```
config-retrieve <SEC>
no config-retrieve
```

Parameters

<SEC> – time interval in seconds, takes values of [1..604800].

Default value

3600

Required privilege level

10

Command mode

CONFIG-ZABBIX-PROXY

Example

```
esr(config-zabbix-proxy)# config-retrieve 200
```

database

This command is used to set the storage location of the zabbix-proxy database.

The use of a negative form (no) of the command sets the default value.

Syntax

```
database <PATH>
```

```
no database
```

Parameters

<PATH> – zabbix-proxy database storage location.

Default value

By default, the zabbix database is stored in the router volatile memory.

Required privilege level

10

Command mode

CONFIG-ZABBIX-PROXY

Example

```
esr(config-zabbix-agent)# database flash:data/zabbix.dat
```

enable

This command enables the functionality of the Zabbix agent.

The use of a negative form (no) of the command disables Zabbix agent functionality.

Syntax

`[no] enable`

Parameters

None.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

Example

```
esr(config-zabbix-agent)# enable
```

hostname

This command sets agent name. Hostname must match the hostname in the Zabbix control panel.

The use of a negative form (no) of the command sets the default agent name.

Syntax

```
hostname <NAME>
no hostname
```

Parameters

<NAME> – system name identifier, set by the string of up to 255 characters.

Default value

Same as the configured hostname of the router.

Required privilege level

10

Command mode

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

Example

```
esr(config-zabbix-agent)# hostname branch_12
```

port

This command sets the TCP port that will be used for Zabbix server requests to the agent.

The use of a negative form (no) of the command removes the value of TCP port.

Syntax

```
port <PORT>
[no] port
```

Parameters

<PORT> – port number, set in the range of [1024..32767].

Default value

None

Required privilege level

10

Command mode

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

Example

```
esr(config-zabbix-agent)# port 20050
```

remote-commands

This command enables the ability to execute commands on the router from the Zabbix server.

The use of a negative form (no) of the command disables the ability to execute commands on the router from the Zabbix server.

Syntax

```
[no] remote-commands
```

Parameters

None.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

Example

```
esr(config-zabbix-agent)# remote-commands
```

server

This command sets the IP address of the Zabbix server for zabbix agent from which incoming connections are allowed.

The use of a negative form (no) of the command removes Zabbix server IP address.

Syntax

```
server <ADDR>
no server
```

Parameters

<ADDR> – server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Default value

None

Required privilege level

10

Command mode

CONFIG-ZABBIX-AGENT

Example

```
esr(config-zabbix-agent)# server 192.168.18.54
```

server

This command sets the IP address and port of the Zabbix server for zabbix proxy from which incoming connections are allowed.

The use of a negative form (no) of the command removes Zabbix server configuration for zabbix-proxy.

Syntax

```
server <ADDR> [port <PORT>]
no server
```

Parameters

<ADDR> – server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<PORT> – UDP port number, set in the range of [1024..32767].

Default value

<ADDR> – none;

<PORT> – 10051.

Required privilege level

10

Command mode

CONFIG-ZABBIX-PROXY

Example

```
esr(config-zabbix-agent)# server 192.168.18.54 port 1285
```

show zabbix-agent configuration

This command displays the Zabbix agent configuration.

Syntax

```
show zabbix-agent configuration
```

Parameters

None.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show zabbix-agent configuration
VRF:                            --
State:                          Enabled
Active server:                  --
Active server port:             --
Hostname:                       esr
Port:                           --
Remote commands:                Disabled
Server:                         192.168.18.54
Source address:                 --
Timeout:                        3
```

show zabbix-proxy configuration

This command displays the Zabbix proxy configuration.

Syntax

show zabbix-proxy configuration

Parameters

None.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show zabbix-proxy configuration
VRF:                         --
State:                       Enabled
Hostname:                    ESR-10SLA
Port:                        --
Remote commands:             Enabled
Server:                      192.168.32.228
Server port:                 --
Source address:              --
Timeout:                     3
Configuration retrieve:      30
```

source-address

This command specifies the address from which the connection to the Zabbix server will be established.

The use of a negative form (no) of the command sets the default value.

Syntax

`source-address <ADDR>`

`no source-address`

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

Default value

IP address of the interface from which NTP packet is sent.

Required privilege level

10

Command mode

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

Example

```
esr(config-zabbix-agent)# source-address 192.168.1.57
```

timeout

This command sets the maximum time for processing a Zabbiz-server request.

The use of a negative form (no) of the command sets the default value.

Syntax

```
timeout
no timeout
```

Parameters

<SEC> – time interval in seconds, takes values of [1..30].

Default value

3

Required privilege level

10

Command mode

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

Example

```
esr(config-zabbix-agent)# timeout 20
```

zabbix-agent

This command switches to the configuration mode of the Zabbix agent in global mode or VRF.

The use of a negative form (no) of the command removes the Zabbix agent settings in global mode or VRF.

Syntax

```
[no] zabbix-agent [ vrf <VRF> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# zabbix-agent
```

zabbix-proxy

This command switches to the configuration mode of the Zabbix proxy in global mode or VRF.

The use of a negative form (no) of the command removes the Zabbix proxy settings in global mode or VRF.

Syntax

```
[no] zabbix-proxy [ vrf <VRF> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters;

Default value

None

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# zabbix-proxy
```

29 DHCP Configuration

- DHCP client management
  - ip address dhcp
  - ip dhcp client ignore
  - ip dhcp client lease-time
  - ip dhcp client reboot
  - ip dhcp client retry
  - ip dhcp client select-timeout
  - ip dhcp client timeout
  - ip dhcp client vendor-class-id
  - ip dhcp server address
  - renew dhcp
- IPv6 DHCP client management
  - ipv6 address dhcp
  - ipv6 dhcp client ignore
  - ipv6 dhcp client lease-time
  - ipv6 dhcp client reboot
  - ipv6 dhcp client retry
  - ipv6 dhcp client select-timeout
  - ipv6 dhcp client timeout
  - ipv6 dhcp client vendor-class-id
  - ipv6 dhcp server address
  - renew ipv6 dhcp
- DHCP Relay agent management
  - ip dhcp information option
  - ip dhcp information option action
  - ip dhcp information option format-type access-node-id
  - ip dhcp information option format-type circuit-id
  - ip dhcp information option format-type option
  - ip dhcp information option format-type remote-id
  - ip dhcp information option suboption-type
  - ip dhcp-relay
  - ip helper-address
  - ip helper-address vrrp-group
- IPv6 DHCP Relay agent management
  - ipv6 dhcp-relay
  - ipv6 dhcp-relay interface
- DHCP server monitoring and configuration
  - clear ip dhcp binding
  - address
  - address-range
  - default-lease-time
  - default-router
  - dns-server
  - domain-name
  - excluded-address-range
  - ip dhcp-server
  - ip dhcp-server dscp
  - ip dhcp-server pool
  - ip dhcp-server vendor-class-id
  - max-lease-time
  - netbios-name-server
  - network

    - option
    - show ip dhcp binding
    - show ip dhcp server dscp
    - show ip dhcp server pool
    - show ip dhcp server vendor-specific
    - tftp-server
    - vendor-specific-options
- IPv6 DHCP server monitoring and configuration
    - address
    - address-range
    - default-lease-time
    - dns-server
    - domain-name
    - excluded-address-range
    - ipv6 dhcp-server
    - ipv6 dhcp-server pool
    - ipv6 dhcp-server vendor-class-id
    - max-lease-time
    - network
    - option
    - show ipv6 dhcp binding
    - show ipv6 dhcp server pool
    - show ipv6 dhcp server vendor-specific

DHCP client management

ip address dhcp

This command enables the retrieving of the dynamic IP address of a configurable interface using the DHCP protocol.

The use of a negative form (no) of the command disables retrieving of the dynamic IP address using the DHCP protocol.

Syntax

```
[no] ip address dhcp
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip address dhcp
```

ip dhcp client ignore

This command specifies DHCP options that will be ignored by the client.

Syntax

```
[no] ip dhcp client ignore <OPTION>
```

Parameters

<OPTION> – takes the following values:

- dns-nameserver – DHCP option 6, list of DNS servers;
- domain-name – DHCP option 15, domain name;
- netbios-nameserver – DHCP option 44, list of NetBios servers;
- router – DHCP option 3, default gateway list;
- classless-static-route – DHCP option 121, list of classless static routes;
- classful-static-route – DHCP option 33, list of classless static routes;
- tftp-server-address – DHCP option 66, TFTPserver name (not applicable for bridge interfaces);
- vendor-specific – DHCP option 43, manufacturer specific information.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip dhcp client ignore router
```

ip dhcp client lease-time

This command sets the requested lease time for the network address.

The use of a negative form (no) of the command sets the default value.

Syntax

`ip dhcp client lease-time <TIME>`

`no ip dhcp client lease-time`

Parameters

<TIME> – requested lease time, sets as DD:HH:MM, where:

- DD – days, takes the value of [0..364];
- HH – hours, takes the value of [0..23];
- MM – minutes, takes the value of [0 ..59].

Default value

1 day

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip dhcp client lease-time 00:12:00
```

ip dhcp client reboot

This command sets the time during which the DHCP client will try to get the old IP address before starting to get a new one.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip dhcp client reboot <SEC>
```

```
no ip dhcp client reboot
```

Parameters

<SEC> – time interval in seconds, takes values of [1..600].

Default value

10

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip dhcp client reboot 60
```

ip dhcp client retry

This command sets the interval at which the DHCP client will resume attempts to obtain an IP_address if it has been defined that the DHCP server is not responding.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip dhcp client retry <SEC>
```

```
no ip dhcp client retry
```

Parameters

<SEC> – time interval in seconds, takes values of [1..600].

Default value

300

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip dhcp client retry 180
```

ip dhcp client select-timeout

This command sets the time period during which the DHCP client will choose among the rental offers from the servers, if any. This is used in networks with multiple DHCP servers. Multiple offers may be sent to the client in response to an IP address request. It is possible that one of these sentences is preferable to the other (for example, one sentence may have an address that the client used previously).

The client waits for the specified period of time from the moment of sending the request for receiving the IP address, in case it receives several offers from various DHCP servers. After the specified time, the client accepts one of the offers.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip dhcp client select-timeout <SEC>
no ip dhcp client select-timeout
```

Parameters

<SEC> – time interval in seconds, takes values of [1..600]

Default value

0 seconds – the customer will accept the first offer received.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip dhcp client select-timeout 30
```

ip dhcp client timeout

The command sets the interval after which the client considers DHCP server as unavailable. If there are any rented addresses in the database of client's IP addresses, the lease term of which has not expired, the client will check each of them sequentially and, if it finds the correct one, then the IP address from it will be assigned to the interface. If there are no valid leases in the database, the client will re-request the IP address after the retry interval (dhcp retry) expires. The use of a negative form (no) of the command sets the default value.

Syntax

```
ip dhcp client timeout <SEC>
no ip dhcp client timeout
```

Parameters

<SEC> – time interval in seconds, takes values of [1..600].

Default value

60 seconds

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip dhcp client timeout 300
```

ip dhcp client vendor-class-id

This command sets the value of DHCP Option 60 to get additional settings for DHCP Option 43.

The use of a negative form (no) of the command disables request of this feature.

Syntax

```
ip dhcp client vendor-class-id <NAME>
no ip dhcp client vendor-class-id
```

Parameters

<NAME> – carrier class identifier, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip dhcp client vendor-class-id ELTEX
```

ip dhcp server address

This command sets the IP address of the DHCP server from which the IP address will be requested.

The use of a negative form (no) of the command removes set DHCP server IP address.

Syntax

```
ip dhcp server address <ADDR>
no ip dhcp server address
```

Parameters

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip dhcp server address 10.10.10.1
```

renew dhcp

This command sends a request to update the IP address via DHCP.

Syntax

```
renew dhcp <IF>
```

Parameters

<IF> – name of an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

Required privilege level

10

Command mode

ROOT

Example

```
esr# renew dhcp gigabitethernet 1/0/1
```

IPv6 DHCP client management

ipv6 address dhcp

This command enables the retrieving of the dynamic IPv6 address of a configurable interface using the IPv6 DHCP protocol.

The use of a negative form (no) of the command disables retrieving of the dynamic IPv6 address using the IPv6 DHCP protocol.

Syntax

```
[no] ipv6 address dhcp
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 address dhcp
```

ipv6 dhcp client ignore

This command specifies DHCP options that will be ignored by the client.

Syntax

```
ipv6 dhcp client ignore <OPTION>
```

```
no ipv6 dhcp client ignore
```

Parameters

<OPTION> – takes the following values:

- dns-nameserver – DHCP option 23, list of DNS servers;
- domain-name – DHCP option 24, domain name;
- vendor-specific – DHCP option 17, manufacturer specific information.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 dhcp client ignore dns-nameserver
```

ipv6 dhcp client lease-time

This command sets the requested lease time for the network address.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 dhcp client lease-time <TIME>
```

```
no ipv6 dhcp client lease-time
```

Parameters

<TIME> – requested lease time, sets as DD:HH:MM, where:

- DD – days, takes the value of [0..364];

- HH – hours, takes the value of [0..23];
- MM – minutes, takes the value of [0 ..59].

Default value

1 day

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 dhcp client lease-time 00:12:00
```

ipv6 dhcp client reboot

This command sets the time during which the IPv6 DHCP client will try to get the old IPv6 address before starting to get a new one.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 dhcp client reboot <SEC>
```

```
no ipv6 dhcp client reboot
```

Parameters

<SEC> – time interval in seconds, takes values of [1..600].

Default value

10

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 dhcp client reboot 60
```

ipv6 dhcp client retry

This command sets the interval at which the IPv6 DHCP client will resume attempts to obtain an IPv6_address if it has been defined that the IPv6 DHCP server is not responding.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 dhcp client retry <SEC>
no ipv6 dhcp client retry
```

Parameters

<SEC> – time interval in seconds, takes values of [1..600].

Default value

300

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 dhcp client retry 180
```

ipv6 dhcp client select-timeout

This command sets the time period during which the IPv6 DHCP client will choose among the rental offers from the servers, if any. This is used in networks with multiple IPv6 DHCP servers — in this case, multiple offers may be sent to the client in response to a request for an IPv6 address. It is possible that one of these sentences is preferable to the other (for example, one sentence may have an address that the client used previously).

The client waits for the specified period of time from the moment of sending the request for receiving the IP address, in case it receives several offers from various DHCPv6 servers. After the specified time, the client accepts one of the offers.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ipv6 dhcp client select-timeout <SEC>
```

```
no ipv6 dhcp client select-timeout
```

Parameters

<SEC> – time interval in seconds, takes values of [1..600].

Default value

0 – the customer will accept the first offer received.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 dhcp client select-timeout 30
```

ipv6 dhcp client timeout

The command sets the interval after which the client considers IPv6 DHCP server as unavailable. If there are any rented addresses in the database of client's IPv6 addresses, the lease term of which has not expired, the client will check each of them sequentially and, if it finds the correct one, then the IPv6 address from it will be assigned to the interface. If there are no valid leases in the database, the client will re-request the IPv6 address after the retry interval (dhcp retry) expires.

The use of a negative form (no) of the command sets the default value.

Syntax

`ipv6 dhcp client timeout <SEC>`

`no ipv6 dhcp client timeout`

<SEC> – time interval in seconds, takes values of [1..600].

Default value

60 seconds

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 dhcp client timeout 300
```

ipv6 dhcp client vendor-class-id

This command sets the value of DHCP Option 60 to get additional settings for DHCP Option 43.

The use of a negative form (no) of the command disables request of this feature.

Syntax

`ipv6 dhcp client vendor-class-id <NAME>`

`no ipv6 dhcp client vendor-class-id`

Parameters

<NAME> – carrier class identifier, set by the string of up to 31 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 dhcp client vendor-class-id ELTEX
```

ipv6 dhcp server address

This command sets the IPv6 address of the DHCP server from which the IPv6 address will be requested.

The use of a negative form (no) of the command removes set DHCP server IPv6 address.

Syntax

```
ipv6 dhcp server address <IPV6-ADDR>
no ipv6 dhcp server address
```

Parameters

<IPV6-ADDR> – IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 dhcp server address fc00::1
```

renew ipv6 dhcp

This command sends a request to update the IPv6 address via DHCP.

Syntax

`renew ipv6 dhcp <IF>`

Parameters

<IF> – name of an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

Required privilege level

10

Command mode

ROOT

Example

```
esr# renew ipv6 dhcp gigabitethernet 1/0/1
```

DHCP Relay agent management

ip dhcp information option

This command enables the processing of the option 82 by DHCP Relay agent.

The use of a negative form (no) of the command disables the processing of the option 82 by DHCP Relay agent.

Syntax

`[no] ip dhcp information option`

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp information option
```

ip dhcp information option action

This command allows you to change the decision that will be made by the router upon receipt of a packet with option 82 installed.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip dhcp information option action { replace | keep | drop }
no ip dhcp information option action
```

Parameters

- replace – replace set option 82 by new;
- keep – keep set option 82 without changes;
- drop – remove the package if the installed option 82 is found in it.

Default value

keep

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip dhcp information option action drop
```

ip dhcp information option format-type access-node-id

This command specifies the Access-node-ID in the Circuit ID field of option 82 when using the tr101 format.

The use of a negative form (no) of the command sets the default value.

Syntax

`[no] ip dhcp information option format-type access-node-id <NODE-ID>`

Parameters

<NODE-ID> – Access-node-ID value, sets by the string up to 31 characters.

Default value

Device hostname.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp information option format-type access-node-id R1
```

ip dhcp information option format-type circuit-id

This command sets the value of the Circuit ID field of option 82 when using any format.

The use of a negative form (no) of the command sets the default value.

Syntax

`[no] ip dhcp information option format-type circuit-id <CIRCUIT-ID>`

Parameters

<ID> – network interface identifier, set by the string of up to 63 characters.

Default value

Defined by the format used.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip dhcp information option format-type circuit-id gi1_0_1
```

ip dhcp information option format-type option

This command specifies the port identifier format in the Circuit ID field of option 82 when using the tr101 format.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] ip dhcp information option format-type option <FORMAT> [ delimeter <DELIMITER> ]
```

Parameters

<FORMAT> – port identifier format, takes one of following values:

- sp – string contains slot and port number;
- sv – string contains slot and VLAN number;
- pv – string contains port and VLAN number;
- spv – string contains slot, port and VLAN number;
- bin – binary format: VLAN (4 bytes), slot (1 byte), port (4 bytes).

<DELIMITER> – separator between parameters in a string, can be one of the following values [. , ; # / space].

Default value

Format – spv

Separator ':' for formats: sp, sv, pv

Separator '/' и ':' for format spv

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp information option format-type option spv delimeter #
```

ip dhcp information option format-type remote-id

This command sets the value of the Remote ID field of option 82 when using any format.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] ip dhcp information option format-type remote-id <REMOTE-ID>
```

Parameters

<ID> – network interface identifier, set by the string of up to 63 characters.

Default value

Defined by the format used.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip dhcp information option format-type remote-id R1
```

ip dhcp information option suboption-type

This command selects option 82 format.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] ip dhcp information option suboption-type { tr101 | custom }
```

Parameters

tr101 – recommended TR-101 format of option 82 according to the syntax adopted in TR-101 recommendations (see table 10);

custom – format option 82 in accordance with the format shown in table 11.

Table 10 – Option 82 field format as per TR-101 recommendations

Field	Transmitted information
Circuit ID	Access-Node-Id, default is device hostname (format can be changed using the «ip dhcp information option format-type access-node-id» command. It is described in Section ip dhcp information option format-type access-node-id) String in the form of eth <stacked/slotid/interfaceid>:<vlan> (format can be changed using the «ip dhcp information option format-type option» command. It is described in Section ip dhcp information option format-type option) The last byte is the port number to which the device is connected, sending a dhcp request.
Remote agent ID	Enterprise number – 0089c1 Device Mac Address

Table 11 – Option 82 of custom mode fields format

Field	Transmitted information
Circuit ID	Length (1 byte) Circuit ID type Length (1 byte) VLAN (2 bytes) Module number (1 byte) Port number (4 byte)
Remote agent ID	Length (1 byte) Remote ID Type (1 byte) Length (1 byte) Switch MAC address

Default value

tr101

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp information option suboption-type custom
```

ip dhcp-relay

This command is used to enable DHCP-relay agent.

The use of a negative form (no) of the command disables DHCP-relay agent.

Syntax

```
[no] ip dhcp-relay
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp-relay
```

ip helper-address

This command specifies the IP address of the DHCP server to which DHCP Discover packets intercepted by the DHCP Relay agent will be sent.

The use of a negative form (no) of the command removes the IP address from the list of DHCP servers for the DHCP Relay agent.

Syntax

```
[no] ip helper-address <IP>
```

Parameters

<IP> – DHCP server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. You can specify up to 4 IP addresses separated by commas.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip helper-address 10.10.10.1
```

ip helper-address vrrp-group

This command restricts sending of DHCP Discover packets captured by the DHCP Relay agent if the VRRP group is in the DOWN state.

The use of a negative form (no) of the command allows sending of DHCP Discover packets captured by the DHCP Relay agent if the VRRP group is in the DOWN state.

Syntax

```
ip helper-address vrrp-group <GRID>
no ip helper-address vrrp-group
```

Parameters

<GRID> – VRRP router group identifier, takes values in the range of [1..32].

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ip helper-address vrrp-group 7
```

IPv6 DHCP Relay agent management

ipv6 dhcp-relay

The command enables IPv6 DHCP agent.

The use of a negative form (no) of the command disables IPv6 DHCP agent.

Syntax

```
[no] ipv6 dhcp-relay
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 dhcp-relay
```

ipv6 dhcp-relay interface

This command specifies the outgoing interface and IPv6 address of the DHCP server to which DHCP Solicit packets intercepted by the IPv6 DHCP Relay agent will be sent.

The use of a negative form (no) of the command removes the outgoing interface from the list of interfaces for the IPv6 DHCP Relay agent.

Syntax

```
ipv6 dhcp-relay interface <IF> [ <IPV6-ADDR> ]
no ipv6 dhcp-relay interface <IF>
```

Parameters

<IF> – name of an interface or a group of interfaces is specified in the form described in Section Types and naming order of router interfaces;

<IPV6-ADDR> – DHCP server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF]. If the address is not specified, packets are sending to ff02::1:2. All IPv6 DHCP servers and agents in the local network segment.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# ipv6 dhcp-relay interface gigabitethernet 1/0/2 fc00::1
```

DHCP server monitoring and configuration

clear ip dhcp binding

This command clears the IP addresses issued by the DHCP server.

Syntax

```
clear ip dhcp binding [ <ADDR> ] [vrf <VRF>]
```

Parameters

<ADDR> – IP address, optional parameter, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. If the <ADDR> parameter is specified in the command, then information associated only with the specified address will be displayed.

<VRF> – name of the VRF instance for which the information will be displayed. Set by the string of up to 31 characters.

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear ip dhcp binding
```

address

This command adds an IP address for a specific physical address to the address pool of a configurable DHCP server.

The use of a negative form (no) of the command removes a specified IP address.

Syntax

```
address <ADDR> { mac-address <MAC> | client-identifier <CLIENT-ID> }
no address <ADDR>
```

Parameters

<ADDR> – client IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]; If the command is used for removal, when specifying the 'all' value all IP addresses will be removed.

<MAC> – MAC address of the client, which will be given the IP address, defined as XX: XX: XX: XX: XX: XX where each part takes the values of [00..FF].

<CLIENT-ID> – client identifier according to DHCP option 61.

Required privilege level

10

Command mode

CONFIG-DHCP-SERVER

Example

```
esr(config-dhcp-server)# address 192.168.3.21 mac-address A8:F9:4B:AA:00:40
```

address-range

This command adds a range of IP addresses to the address pool configured by the DHCP server.

The use of a negative form (no) of the command removes a specified range of IP addresses.

Syntax

```
address-range <FROM-ADDR>-<TO-ADDR>
```

```
no address-range { <FROM-ADDR>-<TO-ADDR> | all }
```

Parameters

<FROM-ADDR> – range starting IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<TO-ADDR> – range ending IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

You can specify up to 32 IP addresses separated by commas.

all – remove all configured IP address ranges.

Required privilege level

10

Command mode

CONFIG-DHCP-SERVER

Example

```
esr(config-dhcp-server)# address-range 192.168.3.1-192.168.3.20,192.168.3.24
```

default-lease-time

This command sets the lease time for which the client will receive an IP address if the client has not requested a specific lease time. The use of a negative form (no) of the command sets the default value.

Syntax

```
default-lease-time <TIME>
```

```
no default-lease-time
```

Parameters

<TIME> – IP address lease time, in format DD:HH:MM, where:

- DD – amount of days, takes values of [0..364].
- HH – amount of hours, takes values of [0..23].
- MM – amount of minutes, takes the value of [0 ..59].

Default value

12 hours

Required privilege level

10

Command mode

CONFIG-DHCP-SERVER

Example

```
esr(config-dhcp-server)# default-lease-time 00:04:00
```

default-router

This command specifies a list of default gateway IP addresses that the DHCP server will communicate to clients using DHCP option 3.

The use of a negative form (no) of the command removes a specified addresses from the list of gateways.

Syntax

```
default-router <ADDR>
no default-router { <ADDR> | all }
```

Parameters

<ADDR> – default gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]; You can specify up to 8 IP addresses separated by commas. If the command is used for removal, when specifying the 'all' value all default gateways will be removed.

Required privilege level

10

Command mode

CONFIG-DHCP-SERVER

Example

```
esr(config-dhcp-server)# default-router 192.168.3.1,192.168.3.2
```

dns-server

This command specifies a list of IP addresses of DNS servers. The list is transferred to clients as part of the DHCP options.

The use of a negative form (no) of the command removes DNS server from the list.

Syntax

```
dns-server <ADDR>
```

```
no dns-server { <ADDR> | all }
```

Parameters

<ADDR> – DNS server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. You can specify up to 8 IP addresses separated by commas. If the command is used for removal, when specifying the 'all' value all DNS servers will be removed.

Required privilege level

10

Command mode

CONFIG-DHCP-SERVER

Example

```
esr(config-dhcp-server)# dns-server 8.8.8.8,8.8.4.4
```

domain-name

This command specifies the DNS name of the network domain. The domain name is transferred to clients as part of the DHCP option 15.

The use of a negative form (no) of the command removes a specified domain name.

Syntax

```
domain-name <NAME>
```

```
no domain-name
```

Parameters

<NAME> – router domain name, set by the string from 1 to 255 characters.

Required privilege level

10

Command mode

CONFIG-DHCP-SERVER

Example

```
esr(config-dhcp-server)# domain-name eltex.loc
```

excluded-address-range

This command excludes a range of IP addresses from the address pool configured by the DHCP server.

The use of a negative form (no) of the command removes a specified range of IP addresses.

Syntax

```
[no] excluded-address-range <FROM-ADDR>-<TO-ADDR>
```

Parameters

<FROM-ADDR> – range starting IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<TO-ADDR> – range ending IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

You can specify up to 32 IP addresses separated by commas.

Required privilege level

10

Command mode

CONFIG-DHCP-SERVER

Example

```
esr(config-dhcp-server)# excluded-address-range 192.168.3.1-192.168.3.20,192.168.3.24
```

ip dhcp-server

This command enables DHCP server.

The use of a negative form (no) of the command disables DHCP server.

Syntax

```
[no] ip dhcp-server [vrf <VRF>]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the NTP server will operate. Set by the string of up to 31 characters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp-server
```

ip dhcp-server dscp

The command sets the DSCP code value for the use in IP headers of DHCP server outgoing packets.

The use of a negative form (no) of the command sets the default DSCP value.

Syntax

```
ip dhcp-server dscp <DSCP>
no ip dhcp-server dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Default value

61

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp-server dscp 40
```

ip dhcp-server pool

This command creates a pool of IP addresses of the DHCP server and switch to its configuration mode.

The use of a negative form (no) of the command removes a specified pool of IP addresses.

Syntax

```
ip dhcp-server pool <NAME> [vrf <VRF>]
no ip dhcp-server pool { <NAME> | all }
```

Parameters

<NAME> – NAT addresses pool name, set by the string of up to 31 characters. The use of a negative form (no) of the command with ‘all’ parameter removes all IP address pools. If the command is used for removal, when specifying the 'all' value all IP addresses will be removed.

<VRF> – VRF instance name, within which the NTP server will operate. Set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp-server pool lan
```

ip dhcp-server vendor-class-id

This command creates a provider class identifier (DHCP Option 60) and switch to its configuration mode.

The use of a negative form (no) of the command removes a specified supplier class identifier.

Syntax

```
[no] ip dhcp-server vendor-class-id <NAME>
```

Parameters

<NAME> – carrier class identifier, set by the string of up to 31 characters. The use of a negative form (no) of the command with 'all' parameter removes all manufacturer class identifiers.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip dhcp-server vendor-class-id ELTEX
```

max-lease-time

This command sets the maximum lease time for IP addresses. If the DHCP client requests a lease time that exceeds the maximum value, the time specified by this command will be set. The use of a negative form (no) of the command sets the default value.

Syntax

```
max-lease-time <TIME>
```

```
no max-lease-time
```

Parameters

<TIME> – maximal IP address lease time, sets in format DD:HH:MM, where:

- DD – amount of days, takes values of [0..364].
- HH – amount of hours, takes values of [0..23].
- MM – amount of minutes, takes the value of [0 ..59].

Default value

1 day

Required privilege level

10

Command mode

CONFIG-DHCP-SERVER

Example

```
esr(config-dhcp-server)# max-lease-time 00:16:00
```

netbios-name-server

This command configures the 44 DHCP option (sets the IP address of the NetBIOS server).

The use of a negative form (no) of the command disables the transfer of the IP address of the NetBIOS server (option 44).

Syntax

```
[no] netbios-name-server <ADDR>
```

Parameters

<ADDR> – NetBIOS server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. You can set up to 4 IP addresses. The use of a negative form (no) of the command with ‘all’ parameter removes all IP addresses.

Required privilege level

10

Command mode

CONFIG-DHCP-SERVER

Example

```
esr(config-dhcp-server)# netbios-name-server 192.168.45.15
```

network

This command sets the IP address and mask for the subnet from which the pool of IP addresses will be allocated.

The use of a negative form (no) of the command removes subnet settings in pool.

Syntax

```
network <ADDR/LEN>
```

```
no network
```

Parameters

<ADDR/LEN> – IP subnet, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32].

Required privilege level

10

Command mode

CONFIG-DHCP-SERVER

Example

```
esr(config-dhcp-server)# network 192.168.3.0/24
```

option

This command specifies DHCP options.

The use of a negative form (no) of the command removes set option.

Syntax

```
option <CODE> { ip-address <ADDR> | ascii-text <STRING> | hex-bytes <HEX> }
```

```
no option { <CODE> | all }
```

Parameters

<CODE> – DHCP option code, set in the form of a number [0..255].

<HEX> – vendor-specific information, specified in hexadecimal format up to 255 bytes.

<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. You can specify up to 15 IP addresses separated by commas.

<URL> – reference address, set by the string from 1 to 255 characters.

all – removes all options.

Default value

None

Required privilege level

10

Command mode

CONFIG-DHCP-SERVER

Example

```
esr(config-dhcp-server)# option 43 ip-address 10.10.1.1
```

show ip dhcp binding

This command displays the IP addresses issued by the DHCP server.

Syntax

```
show ip dhcp binding [ <ADDR> ] [vrf <VRF>]
```

Parameters

<ADDR> – IP address, optional parameter, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. If the <ADDR> parameter is specified in the command, then information associated only with the specified address will be displayed.

<VRF> – name of the VRF instance for which the information will be displayed. Set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip dhcp binding
Allocated          MAC address          Binding    Lease expires at
address                                 type
----------------   ------------------   --------   -----------------------------
192.168.1.3        50:46:5d:a5:3f:91    dynamic    Thursday 2014/01/01 12:42:12
```

show ip dhcp server dscp

This command displays the DSCP value for DHCP messages.

Syntax

```
show ip dhcp server dscp
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr # show ip dhcp server dscp
DSCP:    32
```

show ip dhcp server pool

This command displays configured IP address pools. When specifying a name, only information for the specified pool is displayed.

Syntax

```
show ip dhcp server pool [ <POOL_NAME> ]
```

Parameters

<POOL_NAME> – pool name, optional parameter.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip dhcp server pool lan-pool
name:               lan-pool
network:            192.168.1.0/24
address-ranges:     192.168.1.2-192.168.1.254
default-router:     192.168.1.1
max lease time:     1:0:0 (day:hour:min)
default lease time: 0:12:0 (day:hour:min)
```

show ip dhcp server vendor-specific

This command displays configured DHCP options 43 and 60.

Syntax

```
show ip dhcp server vendor-specific
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip dhcp server vendor-specific
Vendor ID          Vendor options
----------------   --------------------------------
 ELTEX             0x0b0931302e312e39302e32
```

tftp-server

This command sets TFTP server IP address. The address is transferred to clients as part of DHCP option 150.

The use of a negative form (no) of the command removes set TFTP server IP address.

Syntax

```
tftp-server <ADDR>
no tftp-server
```

Parameters

<ADDR> – TFTP server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

config-dhcp-server

Example

```
esr(config-ipv6-dhcp-server)# tftp-server 192.168.0.1
```

vendor-specific-options

This command specifies vendor-specific information (DHCP Option 43).

The use of a negative form (no) of the command removes a specific carrier information.

Syntax

```
vendor-specific-options <HEX>
no vendor-specific-options
```

Parameters

<HEX> – vendor-specific information, specified in hexadecimal format up to 128 symbols.

Required privilege level

10

Command mode

CONFIG-DHCP-VENDOR-ID

Example

```
esr(config-dhcp-vendor-id)# vendor-specific-options 0b0931302e312e39302e320
```

IPv6 DHCP server monitoring and configuration

address

This command adds an IPv6 address for a specific physical address to the address pool of a configurable DHCP server.

The use of a negative form (no) of the command removes a specified IPv6 address.

Syntax

```
address <IPV6-ADDR> mac-address <MAC>
```

```
no address <IPV6-ADDR>
```

Parameters

<IPV6-ADDR> – client IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF]. If the command is used for removal, when specifying the 'all' value all IPv6 addresses will be removed.

<MAC> – MAC address of the client, which will be given the IP address, defined as XX: XX: XX: XX: XX: XX where each part takes the values of [00..FF].

Required privilege level

10

Command mode

CONFIG-IPV6-DHCP-SERVER

Example

```
esr(config-ipv6-dhcp-server)# address fc00::2 mac-address A8:F9:4B:AA:00:40
```

address-range

This command adds a range of IPv6 addresses to the address pool configured by the DHCP server.

The use of a negative form (no) of the command removes a specified range of IPv6 addresses.

Syntax

```
[no] address-range <FROM-ADDR>-<TO-ADDR>
```

```
no address-range all
```

Parameters

<IPV6-ADDR> – server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<IPV6-ADDR> – server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

You can specify up to 32 IP addresses separated by commas.

all – remove all configured IPv6 address ranges.

Required privilege level

10

Command mode

CONFIG-IPV6-DHCP-SERVER

Example

```
esr(config-ipv6-dhcp-server)# address-range fc00::1-fc00:12,fc00::15-fc00::25
```

default-lease-time

This command sets the lease time for which the client will receive an IPv6 address if the client has not requested a specific lease time.

The use of a negative form (no) of the command sets the default value.

Syntax

```
default-lease-time <TIME>
```

```
no default-lease-time
```

Parameters

<TIME> – IP address lease time, in format DD:HH:MM, where:

- DD – amount of days, takes values of [0..364].
- HH – amount of hours, takes values of [0..23].
- MM – amount of minutes, takes the value of [0 ..59].

Default value

12 hours

Required privilege level

1

Command mode

CONFIG-IPV6-DHCP-SERVER

Example

```
esr(config-ipv6-dhcp-server)# default-lease-time 00:04:00
```

dns-server

This command specifies a list of IPv6 addresses of DNS servers. The list is transferred to clients as part of the DHCP options.

The use of a negative form (no) of the command removes DNS server from the list.

Syntax

```
dns-server <IPV6-ADDR>
```

```
no dns-server { <IPV6-ADDR> | all }
```

Parameters

<IPV6-ADDR> – DNS server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF]. You can specify up to 8 IP addresses separated by commas. If the command is used for removal, when specifying the 'all' value all DNS servers will be removed.

Required privilege level

10

Command mode

CONFIG-IPV6-DHCP-SERVER

Example

```
esr(config-ipv6-dhcp-server)# dns-server 8.8.8.8,8.8.4.4
```

domain-name

This command specifies the name of the network domain. The domain name is transferred to clients as part of the DHCP options.

The use of a negative form (no) of the command removes a specified domain name.

Syntax

```
domain-name <NAME>
```

```
no domain-name
```

Parameters

<NAME> – router domain name, set by the string from 1 to 255 characters.

Required privilege level

10

Command mode

CONFIG-IPV6-DHCP-SERVER

Example

```
esr(config-ipv6-dhcp-server)# domain-name eltex.loc
```

excluded-address-range

This command excludes a range of IPv6 addresses from the address pool configured by the DHCP server.

The use of a negative form (no) of the command removes a specified range of IPv6 addresses.

Syntax

```
[no] excluded-address-range <FROM-ADDR>-<TO-ADDR>
```

Parameters

<IPV6-ADDR> – server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

<IPV6-ADDR> – server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF];

You can specify up to 32 IP addresses separated by commas.

Required privilege level

10

Command mode

CONFIG-IPV6-DHCP-SERVER

Example

```
esr(config-ipv6-dhcp-server)# excluded-address-range 10::1-10::9
```

ipv6 dhcp-server

The command enables IPv6 DHCP server.

The use of a negative form (no) of the command disables IPv6 DHCP server.

Syntax

```
[no] ipv6 dhcp-server [ vrf <VRF> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the NTP server will operate.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 dhcp-server
```

ipv6 dhcp-server pool

This command creates a pool of IPv6 addresses of the DHCP server and switch to its configuration mode.

The use of a negative form (no) of the command removes a specified IPv6 address pool.

Syntax

```
[no] ipv6 dhcp-server pool <NAME> [vrf <VRF>]
```

Parameters

<NAME> – IPv6 server profile name, set by the string of up to 31 characters. The use of a negative form (no) of the command with 'all' parameter removes all IPv6 address pools.

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the NTP server will operate.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 dhcp-server pool lan
```

ipv6 dhcp-server vendor-class-id

This command creates a provider class identifier (DHCP Option 60) and switch to its configuration mode.

The use of a negative form (no) of the command removes a specified supplier class identifier.

Syntax

```
[no] ipv6 dhcp-server vendor-class-id <NAME>
```

Parameters

<NAME> – carrier class identifier, set by the string of up to 31 characters. The use of a negative form (no) of the command with 'all' parameter removes all manufacturer class identifiers.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ipv6 dhcp-server vendor-class-id ELTEX
```

max-lease-time

This command sets the maximum lease time for IPv6 addresses. If the IPv6 DHCP client requests a lease time that exceeds the maximum value, the time specified by this command will be set. The use of a negative form (no) of the command sets the default value.

Syntax

```
max-lease-time <TIME>
no max-lease-time
```

Parameters

<TIME> – maximal IP address lease time, sets in format DD:HH:MM, where:

- DD – amount of days, takes values of [0..364].
- HH – amount of hours, takes values of [0..23].
- MM – amount of minutes, takes the value of [0 ..59].

Default value

1 day

Required privilege level

10

Command mode

CONFIG-IPV6-DHCP-SERVER

Example

```
esr(config-ipv6-dhcp-server)# max-lease-time 00:16:00
```

network

This command sets the IPv6 address and mask for the subnet from which the pool of IP addresses will be allocated.

The use of a negative form (no) of the command removes subnet settings in pool.

Syntax

```
network <IPV6-ADDR/LEN>
no network
```

Parameters

<IPV6-ADDR/LEN> – IP address and prefix of a subnet, defined as X:X:X:X::X/EE where each X part takes values in hexadecimal format [0..FFFF] and EE takes values of [1..128].

Required privilege level

10

Command mode

CONFIG-IPV6-DHCP-SERVER

Example

```
esr(config-ipv6-dhcp-server)# network fc00::/110
```

option

This command specifies DHCP options.

The use of a negative form (no) of the command removes this option.

Syntax

```
option <CODE> { ip-address <IPV6-ADDR> | ascii-text <STRING> | hex-bytes <HEX> }
no option { <CODE> | all }
```

Parameters

<CODE> – DHCP option code, set in the form of a number [1..255].

<HEX> – vendor-specific information, specified in hexadecimal format up to 255 symbols.

<IPV6-ADDR> – IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF]. You can specify up to 8 IP addresses separated by commas.

<URL> – reference address, set by the string from 1 to 255 characters.

all – removes all options.

Default value

None

Required privilege level

10

Command mode

CONFIG-IPV6-DHCP-SERVER

Example

```
esr(config-ipv6-dhcp-server)# option 43 ip-address fc00::/110
```

show ipv6 dhcp binding

This command displays the IPv6 addresses issued by the DHCP server.

Syntax

```
show ipv6 dhcp binding [ <IPV6-ADDR> ]
```

Parameters

<IPV6-ADDR> – DHCP server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF]. If the <IPV6-ADDR> parameter is specified in the command, then information associated only with the specified address will be displayed.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 dhcp binding
IPv6 address                                          Lease expires at
---------------------------------------------------   ---------------------------
2001:db8:0:1::18                                      Tuesday 2015/06/09 23:39:45
```

show ipv6 dhcp server pool

This command displays configured IPv6 address pools. When specifying a name, only information for the specified pool is displayed.

Syntax

`show ipv6 dhcp server pool [ <POOL_NAME> ]`

Parameters

<NAME> – NAT addresses pool name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 dhcp server pool lan-pool-ipv6
name:                lan-pool-ipv6
network:             fc00::/110
domain name:         example.com
address-ranges:      fc00::1-fc00::2
                            fc00::3
addresses:           --
dns-server:          fc00:2::3
max lease time:      001:00:00 (day:hour:min)
default lease time:  000:12:00 (day:hour:min)
```

show ipv6 dhcp server vendor-specific

This command displays configured DHCP options 43 and 60.

Syntax

```
show ipv6 dhcp server vendor-specific
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ipv6 dhcp server vendor-specific
Vendor ID          Vendor options
----------------   --------------------------------
 ELTEX             0x0b0931302e312e39302e32
```

30 SLA configuration

- General SLA configuration commands
  - enable
  - frequency
  - ip sla
  - ip sla logging
  - ip sla logging level
  - ip sla mode
  - ip sla responder cisco
  - ip sla responder cisco port
  - ip sla test
  - ip sla schedule
  - packet-size
  - show ip sla configuration
  - show ip sla test configuration
  - show ip sla test statistics
  - thresholds delay
  - thresholds jitter
  - thresholds losses
- Eltex-SLA configuration
  - authentication algorithm
  - authentication key-chain
  - authentication key-string
  - control-phase authentication algorithm
  - control-phase authentication key-id
  - control-phase authentication key-string
  - control-phase destination-port
  - control-phase source-port
  - control-phase timeout
  - cos
  - dscp
  - history
  - icmp-echo
  - ip sla key-chain
  - ip sla responder
  - ip sla responder eltex
  - ip sla responder eltex port
  - timeout
  - timeout
  - ttl
  - udp-jitter
  - vrf
- SLA Weelink configuration (service quality monitoring system)
  - ip sla hostname
  - ip sla portal
  - ip sla responder udp ports
  - ip sla responder udp ports
  - udp-jitter

General SLA configuration commands

enable

This command enables the SLA test.

The use of a negative form (no) of the command disables SLA test.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters

Default value

Off

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# enable
```

frequency

This command sets the time between starts of the SLA session.

The use of a negative form (no) of the command sets the default value.

Syntax

```
frequency <TIME>
no frequency
```

Parameters

<TIME> – time, takes value in seconds [1..604800]

Default value

10

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# frequency 3600
```

ip sla

This command enables the IP-SLA (SLA-agent) functionality.

The use of a negative form (no) of the command disables IP-SLA functionality.

Syntax

```
[no] ip sla
```

Parameters

None

Default value

Disabled

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip sla
```

ip sla logging

This command enables logging of SLA events.

The use of a negative form (no) of the command disables the limitations.

Syntax

`[no] ip sla logging`

Parameters

The command does not contain parameters

Default value

Disabled

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip sla logging
```

ip sla logging level

This command sets the message severity level for logging IP-SLA events.

The use of a negative form (no) of the command sets the default value.

Syntax

`ip sla logging level <SEVERITY>`

`no ip sla logging level`

Parameters

<SEVERITY> – message importance level, takes values (in order of decreasing importance):

- error – error messages;
- warning – warnings, non-emergency messages;
- notice – messages about important system events;
- information – system information messages;
- debug – debugging messages provide the user with information to correctly configure the system;
- trace – full information about all events.

Default value

info

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip sla logging level warning
```

ip sla mode

This command sets the operation mode of the SLA-agent.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip sla mode <MODE>
no ip sla mode
```

Parameters

<MODE> – SLA-agent operation mode. Takes the following values:

eltex – SLA-agent operation in the mode developed by Eltex;

wisla-local – SLA-agent operation in the mode developed by Wellink;

wisla-remote – SLA-client operation in the mode developed by Wellink.

Default value

eltex

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip sla mode wisla
```

ip sla responder cisco

This command activates the SLA-responder functionality for the Cisco-SLA-agent on the interface.

The use of a negative form (no) of the command to deactivate the SLA-responder functionality for Cisco-SLA-agent on the interface.

Syntax

```
[no] ip sla responder cisco
```

Parameters

None

Default value

Disabled

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP

CONFIG-L2TP

CONFIG-PPTP

CONFIG-VTI

CONFIG-E1

CONFIG-MULTILINK

Example

```
esr(config-if-gi)# ip sla responder cisco
```

ip sla responder cisco port

This command sets the UDP port on which to listen for requests from the Cisco-SLA-agent.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip sla responder cisco port <PORT>
no ip sla responder cisco port
```

Parameters

<PORT> – UDP port number in the range of [1..65535].

Default value

Not limited

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP

CONFIG-L2TP

CONFIG-PPTP

CONFIG-VTI

CONFIG-E1

CONFIG-MULTILINK

Example

```
esr(config-if-gi)# ip sla responder cisco port 7777
```

ip sla test

This command is used to create an SLA-test (SLA-agent) and switch to its configuration mode.

The use of a negative form (no) of the command removes specified SLA test.

Syntax

```
[no] ip sla test <NUM>
```

Parameters

<NUM> – number of SLA test in the range [1..10000].

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip sla test 10
```

ip sla schedule

This command sets the schedule of the SLA tests. If the SLA test is created and activated, you need to create this schedule for it.

The use of a negative form (no) of the command removes SLA test schedule.

Syntax

```
ip sla schedule <TEST-NUMBER> [ life { <LIFE-TIME> | forever } ] [ start-time { <MONTH>
<DAY> <TIME> | now } ]
```

```
[no] ip sla schedule <TEST-NUMBER>
```

Parameters

<TIME> – time in seconds, takes values of [1..10000].

<TIME> – time in seconds, takes values of [1..2147483647].

forever – test lifetime is not limited.

<TIME> – system timer, defined as HH:MM:SS, where:

- HH – hours, takes the value of [0..23];
- MM – minutes, takes the value of [0..59];
- SS – seconds, takes the value of [0..59].

<MONTH> – month, takes the following values [January/February/March/April/May/June/July/August/September/October/November/December];

<DAY> – day of the month, takes values of [1..31];

now – start test immediately.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip sla schedule 1 forever now
```

packet-size

This command sets the size of outgoing packets to the SLA-responder.

The use of a negative form (no) of the command sets the default value.

Syntax

```
packet-size <SIZE>
no packet-size
```

Parameters

<SIZE> – SLA test packets size, takes values in range [64..10000].

Default value

74

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# packet-size 256
```

show ip sla configuration

This command displays information about the current SLA-agent settings.

Syntax

```
show ip sla configuration
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show ip sla configuration
Mode:                        Eltex
State:                       Enabled
Logging:                     information
```

show ip sla test configuration

This command displays information about configured SLA tests.

Syntax

```
show ip sla test configuration [ <NUM> ] [ vrf <VRF> ]
```

Parameters

<NUM> – number of SLA test in the range [1..10000];

<VRF> – VRF name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# sh ip sla test configuration 1
Test number:                 1
State:                       Enabled
Control phase:               Disabled
Authentication:              Disabled
Destination address:         1.1.1.1
Destination port:            1000
Frequency:                   10
Interval:                    20
Number of packets:           100
Packet size:                 74
Source address:              192.168.1.100
Source interface:            --
Source port:                 --
DSCP:                        0
COS:                         0
Timeout:                     3000
Number of history records:   10
esr#
```

show ip sla test statistics

This command displays information on the current results of the SLA tests.

Syntax

```
show ip sla test statistics [ <NUM> ] [ vrf <VRF> ]
```

Parameters

<NUM> – number of SLA test in the range [1..10000];

<VRF> – VRF name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# sh ip sla test statistics 1
Test number:                                  1
Transmitted packets:                          100
Lost packets:                                 0 (0%)
Lost packets in forward direction:            0 (0%)
Lost packets in reverse direction:            0 (0%)
One-way delay forward min/avg/max:            0/0/12 milliseconds
One-way delay reverse min/avg/max:            0/0/12 milliseconds
One-way jitter forward min/avg/max:           1/1/1 milliseconds
One-way jitter reverse min/avg/max:           1/1/1 milliseconds
Two-way delay min/avg/max:                    0/0/25 milliseconds
Duplicate packets:                            0
Out of sequence packets in forward direction: 0
Out of sequence packets in reverse direction: 0
```

thresholds delay

This command sets the thresholds for triggering alerts about exceeding (high) or normalizing (low) the values of delay.

The use of the negative form of the command (no) removes thresholds and disables delay parameter tracking.

Syntax

```
thresholds delay { high <DELAY> | low <DELAY> | forward { high <DELAY> | low <DELAY> } | reverse {high <DELAY> | low <DELAY> } }
```

```
no thresholds delay { high | low | forward { high | low } | reverse { high | low } }
```

Parameters

high – alert threshold for exceeding the test parameter value.

low – threshold for triggering alerts about the normalization of the test parameter value.

forward – delay values from SLA-agent to SLA-responder.

reverse – delay values from SLA-responder to SLA-agent.

Without specifying forward/reverse – total delay parameters are set.

<DELAY> – DELAY value, takes values in the range of [1..60000] milliseconds.

Default value

Disabled

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# thresholds delay low 2000
```

thresholds jitter

This command sets the thresholds for triggering alerts about exceeding (high) or normalizing (low) the values of jitter.

The use of the negative form of the command (no) removes thresholds and disables jitter parameter tracking.

Syntax

```
thresholds jitter { high <JITTER> | low <JITTER> | forward { high <JITTER> | low
<JITTER> } | reverse { high <JITTER> | low <JITTER> } }
```

```
no thresholds jitter { high | low | forward { high | low } | reverse { high | low } }
```

Parameters

high – alert threshold for exceeding the test parameter value.

low – threshold for triggering alerts about the normalization of the test parameter value.

forward – jitter values from SLA-agent to SLA-responder.

reverse – jitter values from SLA-responder to SLA-agent.

Without specifying forward/reverse – total jitter parameters are set.

<JITTER> – JITTER value, takes values in the range of [1..60000] milliseconds.

Default value

Disabled

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# thresholds jitter 50000
```

thresholds losses

This command sets the thresholds for triggering alerts about exceeding (high) or normalizing (low) the values of packet loss.

The use of the negative form of the command (no) removes thresholds and disables packet loss parameter tracking.

Syntax

```
thresholds losses { high <NUM-PACKETS> | low <NUM-PACKETS> | forward { high <NUM-PACKETS>
| low <NUM-PACKETS> } | reverse { high <NUM-PACKETS> | low <NUM-PACKETS> } }
```

```
no thresholds losses { high | low | forward | low } | reverse { high | low } }
```

Parameters

high – alert threshold for exceeding the test parameter value.

low – threshold for triggering alerts about the normalization of the test parameter value.

forward – packet loss values from SLA-agent to SLA-responder.

reverse – packet loss values from SLA-responder to SLA-agent.

Without specifying forward/reverse – total amount of packet loss is set.

<NUM-PACKETS> – number of packets, takes values in the range [1..60000] milliseconds.

Default value

Disabled

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# thresholds losses
```

Eltex-SLA configuration

authentication algorithm

This command specifies an algorithm for authenticating incoming requests from the SLA-agent.

The use of the negative form of the command (no) removes an algorithm for authenticating incoming requests from the SLA-agent.

Syntax

```
authentication algorithm <ALGORITHM>
```

```
no authentication algorithm
```

Parameters

<ALGORITHM> – authentication algorithm: sha-256, hmac-sha-256

Default value

Not specified

Required privilege level

15

Command mode

CONFIG-SLA-RESPONDER

Example

```
esr(config-sla-responder)# authentication algorithm sha-256
```

authentication key-chain

This command specifies a set of keys for authenticating incoming requests from the SLA-agent.

The use of the negative form of the command (no) removes a set of keys for authenticating incoming requests from the SLA-agent.

Syntax

```
authentication key-chain <KEYCHAIN>
no authentication key-chain
```

Parameters

<KEYCHAIN> – key list identifier, set by the string of up to 16 characters.

Default value

Not specified

Required privilege level

15

Command mode

CONFIG-SLA-RESPONDER

Example

```
esr(config-sla-responder)# authentication key-chain sla-chain
```

authentication key-string

This command specifies a password for authenticating incoming requests from the SLA-agent.

The use of the negative form of the command (no) removes a password for authenticating incoming requests from the SLA-agent.

Syntax

`authentication key-string ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }`

`no authentication key-string`

Parameters

<CLEAR-TEXT> – password of 8 to 16 bytes;
<ENCRYPTED-TEXT> – encrypted password of 8 to 16 bytes (from 16 to 32 characters) in hexadecimal format (0xYYYY ...) or (YYYY ...).

Default value

None

Required privilege level

15

Command mode

CONFIG-SLA-RESPONDER

Example

```
esr(config-sla-responder)# authentication key-string ascii-text aukey
```

control-phase authentication algorithm

This command sets the authentication algorithm for the SLA test.

The use of a negative form (no) of the command sets the default value.

Syntax

`control-phase authentication algorithm <ALGORITHM>`

`no control-phase authentication algorithm`

Parameters

<ALGORITHM> – hashing algorithm, may take values [sha-256, hmac-sha-256].

Default value

Without encryption

Required privilege level

15

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# control-phase authentication algorithm hmac-sha-256
```

control-phase authentication key-id

This command selects the key number in the key-chain configured by the ip sla key-chain command (see section authentication key-chain), which will be sent for authentication on the SLA-responder.

The use of a negative form (no) of the command removes a number of key.

Syntax

```
control-phase authentication key-id <ID>
no control-phase authentication key-id
```

Parameters

<ID> – key identifier, set in the range of [1..255].

Required privilege level

15

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# control-phase authentication key-id 2
```

control-phase authentication key-string

This command sets the authentication key that will be sent for authentication on the SLA-responder.

The use of the negative form of the command (no) removes the authentication key that will be sent for authentication to the SLA-responder.

Syntax

```
control-phase authentication key-string ascii-text { <CLEAR-TEXT> |encrypted <ENCRYPTED-TEXT>}
```

```
no control-phase authentication key-string
```

Parameters

<CLEAR-TEXT> – password, sets by string from 8 to 16 characters;

<ENCRYPTED-TEXT> – encrypted password from 8 bytes to 16 bytes (16 to 32 characters) in hexadecimal format (0xYYYY...) or (YYYY...).

Required privilege level

15

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# control-phase authentication key-string ascii-text conphkey
```

control-phase destination-port

This command sets the UDP port to which requests for Eltex-SLA-responder will be sent.

The use of a negative form (no) of the command sets the default value.

Syntax

```
control-phase destination-port <PORT>
```

```
no control-phase destination-port
```

Parameters

<PORT> – UDP port number in the range of [1..65535].

Default value

1800

Required privilege level

15

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# control-phase destination-port 9999
```

control-phase source-port

This command sets the UDP port from which requests for Eltex-SLA-responder will be sent.

The use of a negative form (no) of the command sets the default value.

Syntax

```
control-phase source-port <PORT>
no control-phase source-port
```

Parameters

<PORT> – UDP port number in the range of [1..65535].

Default value

None

Required privilege level

15

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# control-phase source-port 3333
```

control-phase timeout

This command sets the time to wait for a response from the SLA-responder, after which the control phase of the SLA test will be considered unsuccessful.

The use of a negative form (no) of the command sets the default value.

Syntax

```
control-phase timeout <TIME>
```

```
no control-phase timeout
```

Parameters

<TIME> – time allotted for the passage of the control phase of the SLA test is specified in milliseconds [1..4294967295].

Default value

3000

Required privilege level

15

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# control-phase timeout 5000
```

cos

This command sets the COS value of outgoing packets to the SLA-responder.

The use of a negative form (no) of the command removes the COS value.

Syntax

```
cos <COS>
```

```
no cos
```

Parameters

<COS> – service classifier in 802.1q packet tag, takes values in the range of [0..7].

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# cos 5
```

dscp

This command sets the DSCP value of outgoing packets to the SLA-responder.

The use of a negative form (no) of the command sets the default value.

Syntax

```
dscp <DSCP>
no dscp
```

Parameters

<DSCP> – DSCP code value, takes values in the range of [0..63].

Default value

0

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# dscp 58
```

history

This command sets the number of stored SLA test results.

The use of a negative form (no) of the command sets the default value.

Syntax

```
history <SIZE>
no history
```

Parameters

<SIZE> – the number of stored results, takes the values [1..10000].

Default value

10

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# history 100
```

icmp-echo

This command sets the icmp mode for testing the communication channel and test parameters.

The use of a negative form (no) of the command removes the configuration of the test mode and parameters.

Syntax

```
icmp-echo <DST-ADDRESS> { source-ip <SRC-ADDRESS> | source-interface { <IF> | <TUN> } }
[ interval <INTERVAL> ] [ num-packets <NUM-PACKETS> ]
```

```
no icmp-echo
```

Parameters

<DST-ADDRESS> – ipv4 address of the SLA-responder. IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<SRC-ADDRESS> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IF> – type and identifier of the interface/tunnel whose IP address will be used as the address of the packet source is specified in the form described in the section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

<INTERVAL> – test packet sending interval, takes the value [1..255] milliseconds;

<NUM-PACKETS> – number of test packets sent within the test is [1..6000];

Default value

<INTERVAL> – 20

<NUM-PACKETS> – 100

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# icmp-echo 10.10.1.1 source-ip 192.168.54.22 num-packets 50 interval 15
```

ip sla key-chain

The command specifies a set of passwords for authentication with SLA-responder/agent via md5 hash algorithm.

The use of a negative form (no) of the command removes a bind to the password set.

Syntax

```
ip sla key-chain <NAME>
no ip sla key-chain
```

Parameters

<NAME> – name, set by the string from 1 to 16 characters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip sla key-chain KEY10
```

ip sla responder

This command switches to the SLA-responder parameter configuration mode.

The use of a negative form of the command (no) removes previously applied SLA-responder parameters.

Syntax

```
[no] ip sla responder [ vrf <VRF> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When this parameter is specified, the SLA-responder is enabled in the specified VRF.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip sla responder
```

ip sla responder eltex

This command activates the SLA-responder functionality for the Eltex-SLA-agent on the interface.

The use of a negative form (no) of the command to deactivate the SLA-responder functionality for Eltex-SLA-agent on the interface.

Syntax

```
[no] ip sla responder eltex
```

Parameters

None

Default value

Disabled

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP

CONFIG-L2TP

CONFIG-PPTP

CONFIG-VTI

CONFIG-E1

CONFIG-MULTILINK

Example

```
esr(config)# ip sla responder eltex
```

ip sla responder eltex port

This command sets the UDP port on which to listen for requests from the Eltex-SLA-agent.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip sla responder eltex port <PORT>
no ip sla responder eltex port
```

Parameters

<PORT> – UDP port number in the range of [1..65535].

Default value

1800

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP

CONFIG-L2TP

CONFIG-PPTP

CONFIG-VTI

CONFIG-E1

CONFIG-MULTILINK

Example

```
esr(config)# ip sla responder eltex port 5555
```

timeout

This command sets the waiting time for the next packet from the SLA-responder, after which the SLA-agent closes the SLA session.

The use of a negative form (no) of the command sets the default value.

Syntax

`timeout <TIME>`

`no timeout`

Parameters

<TIME> – timeout, takes value in milliseconds [1..4294967295].

Default value

3000

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# timeout 1000
```

timeout

This command sets the waiting time for the next packet from the SLA-agent, after which the SLA-responder closes the SLA session.

The use of a negative form (no) of the command sets the default value.

Syntax

`timeout <TIME>`

`no timeout`

Parameters

<TIME> – time allotted for the passage of the SLA test is specified in milliseconds [1..4294967295].

Default value

3000

Required privilege level

10

Command mode

CONFIG-SLA-RESPONDER

Example

```
esr(config-sla-responder)# timeout
```

ttl

This command sets the TTL value of outgoing packets to the SLA-responder.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ttl <TTL>
no ttl
```

Parameters

<TTL> – TTL value, takes values in the range of [1..255].

Default value

64

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# ttl
```

udp-jitter

This command sets the udp mode for testing the communication channel and test parameters.

The use of a negative form (no) of the command removes communication channel testing parameters.

Syntax

```
udp-jitter <DST-ADDRESS> <DST-PORT> { source-ip <SRC-ADDRESS> | source-interface { <IF> | <TUN> } } [ source-port <SRC-PORT> ]  [ interval <INTERVAL> ] [ num-packets <NUM-PACKETS> ]
```

```
no udp-jitter
```

Parameters

<DST-ADDRESS> – ipv4 address of the SLA-responder. IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<DST-PORT> – number of test packets destination UDP port, takes values of [1..65535];

<SRC-ADDRESS> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

<IF> – type and identifier of the interface/tunnel whose IP address will be used as the address of the packet source is specified in the form described in the section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

<SRC-PORT> – number of test packets source UDP port, takes values of [1..65535];

<INTERVAL> – test packet sending interval, takes the value [1..255] milliseconds;

<NUM-PACKETS> – number of test packets sent within the test is [1..6000];

<SRC-ADDRESS> – source address of the test packets of the SLA test is specified by the ipv4 address. May take the value of ipv4-address.

Default value

control enable

<INTERVAL> – 20

<NUM-PACKETS> – 100

<SRC-PORT> – any free

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# udp-jitter 10.10.1.1 50000 source-ip 192.168.54.22 num-packets 50
 interval 15
```

vrf

This command sets the VRF instance in the address space of which the SLA test should operate.

The use of the negative form of the command (no) returns the SLA test to the address space of the global router.

Syntax

`vrf <VRF>`

`no vrf`

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# vrf subrouter
```

SLA Weelink configuration (service quality monitoring system)

⚠ This functionality is activated only with a license

ip sla hostname

This command sets the name of the router for its identification in the wiSLA system.

The use of a negative form (no) of the command sets the default value.

Syntax

`ip sla hostname <NAME>`

`no ip sla hostname`

Parameters

<NAME> – router host name, set by the string of up to 64 characters.

Default value

Same as the hostname of the router

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip sla hostname esr2517
```

ip sla portal

This command sets the name of the router for its identification in the wiSLA system.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip sla portal <URL>
no ip sla portal
```

Parameters

<URL> – reference address, set by the string from 8 to 255 characters.

Default value

Unspecified.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip sla portal wislaurl.loc
```

ip sla responder udp ports

This command enables the SLA responder and specifies the TCP/UDP ports it uses.

The use of a negative form (no) of the command disables the limitations.

Syntax

```
[no] ip sla responder udp ports <PORT-SET-NAME>
```

Parameters

<NAME> – TCP/UDP ports profile name, set by the string of up to 31 characters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip sla responder udp ports SLA_PORTS
```

ip sla responder udp ports

This command restricts the list of UDP ports on which test IP SLA packets can be received.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip sla responder udp ports <OBJECT-GROUP-SERVICE>
no ip sla responder udp ports
```

Parameters

<OBJECT-GROUP-SERVICE> – TCP/UDP ports profile name, set by the string of up to 31 characters.

Default value

No restrictions

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# ip sla responder udp ports OGS_SLA
```

udp-jitter

This command configures the udp jitter test.

The use of a negative form (no) of the command removes udp jitter test.

Syntax

```
[no] udp-jitter <DST-ADDRESS> <DST-PORT> [ control { enable | disable } ] [ interval
<INTERVAL> ] [ num-packets <NUM-PACKETS> ] [ source-ip <SRC-ADDRESS> ] [ source-port
<SRC-PORT> ]
```

Parameters

<DST-ADDRESS> – address of responder SLA test, sets by ipv4 address. May take the value of ipv4-address.

IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<DST-PORT> – number of test packets destination UDP port, takes values of [1..65535].

enable – enable control phase.

disable – disable control phase.

<INTERVAL> – test packet sending interval, takes the value [1..255] milliseconds.

<NUM-PACKETS> – number of test packets sent within the test is [1..6000].

<SRC-ADDRESS> – source address of the test packets of the SLA test is specified by the ipv4 address. May take the value of ipv4-address.

IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<SRC-PORT> – number of test packets source UDP port, takes values of [1..65535].

Default value

control enable

<INTERVAL> – 20

<NUM-PACKETS> – 100

<SRC-ADDRESS> – IP address of the interface through which the SLA responder is available

<SRC-PORT> – any free

Required privilege level

10

Command mode

CONFIG-SLA-TEST

Example

```
esr(config-sla-test)# udp-jitter 10.10.1.1 50000 control enable num-packets 50 interval 15
```

31 Subscriber control configuration (BRAS)

⚠ This functionality is supported only with a license

- aaa das-profile
- aaa services-radius-profile
- aaa sessions-radius-profile
- backup traffic-processing transparent
- bypass-traffic-acl
- class-map
- clear subscriber-control sessions
- default-action
- default-service
- description
- dhcp-option-82-include accept-time
- dhcp-option-82-include enable
- dhcp-option-82-include lease-time
- dhcp-option-82-include size
- enable
- filter-action
- filter-name
- ip proxy http listen-ports
- ip proxy https listen-ports
- ip proxy source-address
- location
- nas-interface
- nas-ip-address
- quota-expired-reauth
- service-subscriber-control
- session accounting
- session ip-authentication
- session l2-roaming disable
- session l2-roaming realtime-accounting
- session mac-authentication
- session unknown-mac-address
- session-timeout
- show subscriber-control configuration
- show subscriber-control radius-servers
- show subscriber-control services count
- show subscriber-control services counters
- show subscriber-control services status
- show subscriber-control sessions count
- show subscriber-control sessions counters
- show subscriber-control sessions status
- subscriber-control
- subscriber-control application-filter
- subscriber-control apps-server-url
- subscriber-control filters-server-url
- subscriber-control peer-address
- subscriber-control thresholds sessions-number
- subscriber-control unused-filters-remove-delay
- thresholds sessions-number
- vrrp-group

aaa das-profile

This command selects a dynamic authorization server (DAS) profile to which CoA requests from the PCRF to change the service policy, as well as requests for operational information from CaptivePortal will be received.

The use of a negative form (no) of the command removes a specified profile of dynamic authorization servers (DAS).

Syntax

```
[no] aaa das-profile <NAME>
```

Parameters

<NAME> – DAS profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# aaa das-profile profile1
```

aaa services-radius-profile

This command selects the profile of RADIUS servers to which requests will be sent to obtain user service parameters. If the profile is not set, the profile will be used 'aaa sessions-radius-profile'.

The use of a negative form (no) of the command removes a specified RADIUS server profile.

Syntax

```
[no] aaa services-radius-profile <NAME>
```

Parameters

<NAME> – RADIUS server profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# aaa services-radius-profile profile1
```

aaa sessions-radius-profile

This command selects the profile of RADIUS servers to which requests will be sent to obtain user session parameters.

The use of a negative form (no) of the command removes a specified RADIUS server profile.

Syntax

```
[no] aaa sessions-radius-profile <NAME>
```

Parameters

<NAME> – RADIUS server profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# aaa sessions-radius-profile profile1
```

backup traffic-processing transparent

This command enables transparent transmission of traffic in the backup state for BRAS.

The use of a negative form (no) of the command disables the transparent transmission of traffic in the backup state for BRAS.

Syntax

```
[no] backup traffic-processing transparent
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# backup traffic-processing transparent
```

bypass-traffic-acl

This command organizes transparent transmission of service traffic (DHCP, DNS, etc.) based on filters.

The use of a negative form (no) of the command disables the transparent traffic transmission.

Syntax

```
bypass-traffic-acl <NAME>
no bypass-traffic-acl
```

Parameters

<NAME> – name of the ACL being bound, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# bypass-traffic-acl LANs
```

class-map

This command binds the specified QoS class to the default service. Passing traffic not included in the QoS class is prohibited.

The use of a negative form (no) of the command removes a bind of class to the service by default.

Syntax

```
[no] class-map <NAME>
```

Parameters

<NAME> – name of the class being bound, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-DEFAULT-SERVICE

Example

```
esr(config-subscriber-default-service)# class-map LAN
```

clear subscriber-control sessions

This command deletes active user control sessions.

Syntax

```
clear subscriber-control sessions [ vrf <VRF> ] [ username <USER-NAME> ] [ session-id
<SESSION-ID> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, active sessions in a specified VRF will be removed;

<NAME> – user name, set by the string of up to 230 characters;

<SESSION-ID> – session identifier, takes values in the range of [1..18446744073709551615].

Required privilege level

10

Command mode

ROOT

Example

```
esr# clear subscriber-control sessions
```

default-action

This command specifies the action to be applied for HTTP/HTTPS packets, URLs (HTTP Host for HTTPS packets) which are not included in the URL list assigned by the «filter-name» command (see section filter-name).

The use of a negative form (no) of the command removes an assigned action.

Syntax

```
default-action <ACT>
```

```
no default-action
```

Parameters

<ACT> – allocated action:

- permit – traffic transfer is permitted;
- deny – traffic transfer is denied.
- redirect <URL> – redirect to the specified URL will be carried out, set by the string of up to 255 characters.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-DEFAULT-SERVICE

Example

```
esr(config-subscriber-default-service)# default-action redirect http://192.162.1.2/cp
```

default-service

Switch to the default service configuration mode. The default service applies to all new user sessions. After authentication, the user is assigned personal services.

Syntax

```
default-service
```

Parameters

The command does not contain parameters.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# default-service
```

description

This command defines the description of the user control profile.

The use of a negative form (no) of the command removes description.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – wan rule description, set by the string of up to 255 characters.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# description "Wi-Fi BRAS"
```

dhcp-option-82-include accept-time

This command sets the time after which an unconfirmed option will be removed.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] dhcp-option-82-include accept-time <SEC>
```

Parameters

<SEC> – time interval in seconds, takes values of [10..3600].

Default value

10 seconds

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# dhcp-option-82-include accept-time 30<>>>>>>>>>>>>>>>>>>>
```

dhcp-option-82-include enable

This command enables a search for option 82 in DHCP packets.

The use of a negative form (no) of the command disables a search for option 82 in DHCP packets.

Syntax

`[no] enable`

Parameters

The command does not contain parameters.

Default value

Process disabled.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# enable
```

dhcp-option-82-include lease-time

This command sets the time after which the record of the confirmed option 82 will be considered outdated.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] dhcp-option-82-include lease-time <SEC>
```

Parameters

<SEC> – time interval in seconds, takes values of [60..86400].

Default value

3600 seconds

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# dhcp-option-82-include lease-time 7200
```

dhcp-option-82-include size

This command sets the maximum table size for storing option 82.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] dhcp-option-82-include size <SIZE>
```

Parameters

<SIZE> – option 82 table size [100..100000].

Default value

100 for ESR-1x

1000 for ESR-20/100/200

10000 for ESR-1000/1200/1510/1511

50000 for ESR-1700/3100

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# dhcp-option-82-include size 25000
```

enable

This command activates the user control profile.

The use of a negative form (no) of the command disables user control profile.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

Process disabled.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# enable
```

filter-action

This command specifies the action to be applied for HTTP/HTTPS packets, URLs (HTTP Host for HTTPS packets) which are included in the URL list assigned by the «filter-name» command (see section filter-name).

The use of a negative form (no) of the command removes an assigned action.

Syntax

```
filter-action <ACT>
no filter-action
```

Parameters

<ACT> – allocated action:

- permit – traffic transfer is permitted;
- deny – traffic transfer is denied.
- redirect <URL> – redirect to the specified URL will be carried out, set by the string of up to 255 characters.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-DEFAULT-SERVICE

Example

```
esr(config-subscriber-default-service)# filter-action redirect http://192.162.1.2/forbidden
```

filter-name

Specify a name of the URL list that will be used to filtrate HTTP/HTTPS traffic of non-authenticated users. The list can be configured locally using the URL profile, or obtained from a remote server (see section subscriber-control application-filter).

The use of a negative form (no) of the command removes a list name.

Syntax

```
filter-name { local <LOCAL-NAME> | remote <REMOTE-NAME> }
```

```
no filter-name
```

Parameters

<LOCAL-NAME> – URL profile name, set by the string of up to 31 characters;

<REMOTE-NAME> – remote server URL list name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-DEFAULT-SERVICE

Example

```
esr(config-subscriber-default-service)# filter-name local BLACK_LIST
```

ip proxy http listen-ports

Define destination TCP ports from which the traffic will be redirected to the router HTTP Proxy server

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip proxy http listen-ports <NAME>
no ip proxy http listen-ports
```

Parameters

<NAME> – TCP/UDP ports profile name, set by the string of up to 31 characters.

Default value

80, 8080

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# ip proxy http listen-ports HTTP_PORTS
```

ip proxy https listen-ports

Define destination TCP ports from which the traffic will be redirected to the router HTTPS Proxy server

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip proxy https listen-ports <NAME>
no ip proxy https listen-ports
```

Parameters

<NAME> – TCP/UDP ports profile name, set by the string of up to 31 characters.

Default value

443, 8443

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# ip proxy https listen-ports HTTPS_PORTS
```

ip proxy source-address

Set router IP address that will be used as source IP address in HTTP/HTTPS packets transmitted by Proxy server

The use of a negative form (no) of the command removes a specified source IP address.

Syntax

```
ip proxy source-address <ADDR>
no ip proxy source-address
```

Parameters

<ADDR> – source IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

Default value

The default is the IP address of the interface from which the packet will be sent.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# ip proxy source-address 10.100.100.2
```

location

This command changes the network interface ID. This identifier is used in HTTP redirects to CaptivePortal, and is also transmitted in the RADIUS account information and when exporting information via the Netflow protocol.

The use of a negative form (no) of the command removes the identifier.

Syntax

```
location <ID>
```

```
no location
```

Parameters

<ID> – network interface identifier, set by the string of up to 220 characters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TPV3

Example

```
esr(config-if-gi)# location “Guest SSID”
```

nas-interface

This command defines router interface IP address of which will be used as the source IP address in packets sent by RADIUS.

The use of a negative form (no) of the command removes a specified source IP address.

Syntax

```
nas-interface {<IF> | <TUN>}
no nas-interface
```

Parameters

<IF> – an interface, specified in the form described in Section Types and naming order of router interfaces;

<TUN> – the name of the tunnel is specified as described in section Types and naming order of router tunnels.

Default value

Not specified.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# nas-interface gi 1/0/1
```

nas-ip-address

This command defines the IP address of the router that will be used as the source IP address in packets sent by RADIUS.

The use of a negative form (no) of the command removes a specified source IP address.

Syntax

```
nas-ip-address <ADDR>
no nas-ip-address
```

Parameters

<ADDR> – source IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

Required privilege level

10

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# nas-ip-address 10.100.100.2
```

quota-expired-reauth

Enable iterative query of quota value when it expires for user services with a configured restriction on the amount of traffic or time Otherwise, after the expiration of the quota, the service will be deactivated, and the user will be assigned the service by default.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] quota-expired-reauth
```

Parameters

The command does not contain parameters.

Default value

When the quota expires, the user will be assigned a default service.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# quota-expired-reauth
```

service-subscriber-control

This command enables user control on the interface. When executed with the 'any' parameter value, user control will work for packets from any subnet, otherwise only for packets from the subnets of the specified IP addresses in the profile.

The use of a negative form (no) of the command disables user control on the interface.

Syntax

```
service-subscriber-control { any | object-group <NAME> }
no service-subscriber-control
```

Parameters

<NAME> – IP addresses profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

Example

```
esr(config-if-gi)# service-subscriber-control object-group LAN
```

session accounting

This command sets the mode of sending RADIUS-accounting messages.

The use of a negative form (no) of the command sets the default value.

Syntax

```
session accounting { all | auth-only }
```

Parameters

all – sending for all sessions;

auth-only – sending for authorized sessions.

Default value

all

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# session accounting all
```

session ip-authentication

Enable session authentication by IP address.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] session IP-authentication
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# session ip-authentication
```

session l2-roaming disable

This command disables transparent roaming of subscribers between L2-interfaces BRAS. In case of changing the L2 interface, the subscriber will need to re-authenticate.

The use of a negative form (no) of the command enables transparent roaming of subscribers between L2-interfaces BRAS.

Syntax

```
[no] session l2-roaming disable
```

Parameters

The command does not contain parameters.

Default value

Enabled transparent roaming of subscribers between L2 interfaces BRAS.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# session l2-roaming disable
```

session l2-roaming realtime-accounting

This command enables the mode of sending RADIUS accounting in real time when changing the L2 interface of the BRAS, which is used to work with the subscriber.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] session l2-roaming realtime-accounting
```

Parameters

The command does not contain parameters.

Default value

RADIUS accounting with a modified L2 interface is sent after Interim-Update.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# session l2-roaming realtime-accounting
```

session mac-authentication

Enable session authentication by MAC address.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] session mac-authentication
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# session mac-authentication
```

session unknown-mac-address

This command denies the passage of traffic in an authenticated session for packets that have changed the source MAC address since the user’s authentication. Also, when receiving a packet with a different source MAC address, a message will appear in the SYSLOG.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] session unknown-mac-address filtering
```

Parameters

The command does not contain parameters.

Default value

Passing traffic with a different source MAC address allowed.

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# session unknown-mac-address
```

session-timeout

Specify the interval after which, if a user has not sent any packets, the session is considered to be outdated and is removed from the device.

The use of a negative form (no) of the command sets the default value.

Syntax

```
session-timeout <SEC>
no session-timeout
```

Parameters

<SEC> – time interval in seconds, takes values of [120..3600].

Default value

120 seconds

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-DEFAULT-SERVICE

Example

```
esr(config-subscriber-default-service)# session-timeout 155
```

show subscriber-control configuration

The command displays user control configuration parameters.

Syntax

```
show subscriber-control configuration [ vrf <VRF> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show subscriber-control configuration
State:                Enabled
Description:          --
NAS IP address:       192.168.107.201
Sessions radius profile:       RADIUS
Services radius profile:       --
DAS profile:          bras2
Quota expired reauth: Disabled
Default service:
 Class map:             list1
 Filter name:           defaultserv
 Filter type:           local
 Filter action:         permit
 Default action:        redirect
 Default redirect URL: http://192.168.107.213:8080/eltex_portal/
```

show subscriber-control radius-servers

The command displays information about used RADIUS servers.

Syntax

```
show subscriber-control radius-servers [ vrf <VRF> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, NTP configuration will be displayed in a specified VRF.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show subscriber-control radius-servers
IP address         Port     VRF              Usage            Connections count   Dead interval
Dead time
----------------   ------   -------------    -------------    -----------------   -------------
---------
172.16.0.134       31813    --               services acct    0                   10
--
172.16.0.134       31812    --               services auth    0                   10
--
172.16.0.135       31813    --               sessions acct    0                   10
--
172.16.0.135       31812    --               sessions auth    0                   10
--
```

show subscriber-control services count

These commands display information and statistics on user control session services.

Syntax

```
show subscriber-control services count [ { vrf <VRF> | service <SERVICE-NAME> } ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, NTP configuration will be displayed in a specified VRF.

<NAME> – host name, set by the string of up to 220 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show subscriber-control services count
Service           Active        Not active
---------------   -----------   -----------
INTERNET          17            0
WELCOME           0             2
```

show subscriber-control services counters

These commands display counters for user control session services.

Syntax

```
show subscriber-control services counters session-id <SESSION-ID> [ vrf <VRF> ]
[ service-id <SERVICE-ID> ] [ service <SERVICE-NAME> ]
```

Parameters

counters – view statistics on the service user;

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, NTP configuration will be displayed in a specified VRF.

<SESSION-ID> – session identifier, takes values in the range of [1..18446744073709551615].

<SERVICE-ID> – session identifier, takes values in the range of [1..18446744073709551615].

<NAME> – host name, set by the string of up to 220 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show subscriber-control services status counters session-id 5116089176692896081
Service id               Service name       Recv packets      Recv bytes        Send packets
Send bytes
--------------------     ---------------    --------------    --------------    --------------
--------------
5152117973711859021      A INTERNET         2018              1615309           1761
281017
```

show subscriber-control services status

These commands display perative information on user control session services.

Syntax

```
show subscriber-control services status [ session-id <SESSION-ID> ] [ { service-id
<SERVICE-ID> | service <SERVICE-NAME> ] [ vrf <VRF> ]
```

Parameters

status – view operational information on the service user;

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, NTP configuration will be displayed in a specified VRF.

<SESSION-ID> – session identifier, takes values in the range of [1..18446744073709551615].

<SERVICE-ID> – session identifier, takes values in the range of [1..18446744073709551615].

<NAME> – host name, set by the string of up to 220 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show subscriber-control services status
Service id    Session id    Service name    User name    Quota volume    Quota time
                                                         (Bytes)         (Seconds)
------------  ------------  --------------  -----------  --------------  --------------
2522015791    2161727821    INTERNET5       79001110011  --              --
```

show subscriber-control sessions count

These commands display information and statistics on user control sessions.

Syntax

```
show subscriber-control sessions count [ vrf <VRF> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, active sessions will be displayed in a specified VRF.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show subscriber-control sessions count
All:                18
Authenticated:      17
Not authenticated: 1
```

show subscriber-control sessions counters

These commands display counters for user session.

Syntax

```
show subscriber-control sessions counters { session-id <SESSION-ID> | username <SERVICE-
NAME> } [ vrf <VRF> ]
```

Parameters

status – view operational information on the user session;

counters – view statistics on the user session;

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, active sessions will be displayed in a specified VRF.

<SESSION-ID> – session identifier, takes values in the range of [1..18446744073709551615];

<NAME> – user name, set by the string of up to 230 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show subscriber-control sessions counters session-id 5116089176692896081
User name          Recv packets      Recv bytes        Send packets      Send bytes
---------------    --------------    --------------    --------------    --------------
79231552525        2198              1697762           1984              313452
```

show subscriber-control sessions status

These commands display operative information for user session.

Syntax

```
show subscriber-control sessions status [ { session-id <SESSION-ID> | username <SERVICE-NAME> } ] [ vrf <VRF> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters. When specifying this parameter, active sessions will be displayed in a specified VRF.

<SESSION-ID> – session identifier, takes values in the range of [1..18446744073709551615];

<NAME> – user name, set by the string of up to 230 characters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show subscriber-control sessions status
Session id    User name    IP address       MAC address         Interface   Domain
------------  -----------  ------------     -----------------   ---------   ---------
2161727821    79001110011  192.168.244.12   c4:12:f5:d4:af:70   bridge 13   root
esr# show subscriber-control sessions counters session-id 2161727821
User name         Recv packets     Recv bytes       Send packets     Send bytes
---------------   --------------   --------------   --------------   --------------
79001110011       243              87056            294              35961
```

subscriber-control

This command creates a user control profile and switch to its configuration mode.

The use of a negative form (no) of the command removes a specified user control profile.

Syntax

```
[no] subscriber-control [ vrf <VRF> ]
```

Parameters

<VRF> – VRF instance name, set by the string of up to 31 characters, within which the user control will operate.

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# subscriber-control
```

subscriber-control application-filter

This command enables application control on the interface.

The use of a negative form (no) of the command disables application control on the interface.

Syntax

```
subscriber-control application-filter <NAME>
no subscriber-control application-filter
```

Parameters

<NAME> – application profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-GI

CONFIG-TE

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TPV3

Example

```
esr(config-if-gi)# subscriber-control application-filter LIST
```

subscriber-control apps-server-url

Specify URL address of the server providing lists of traffic filtration applications Lists are requested from the server at the time of user authentication.

The use of a negative form (no) of the command removes a specified server URL.

Syntax

```
subscriber-control apps-server-url <URL>
```

```
no subscriber-control apps-server-url
```

Parameters

<URL> – reference address, set by the string from 8 to 255 characters.

Default value

None

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# subscriber-control apps-server-url "http://192.168.1.1/files/"
```

subscriber-control filters-server-url

This command sets the address of the server that provides URL lists for filtering HTTP/HTTPS traffic. Lists are requested from the server at the time of user authentication.

The use of a negative form (no) of the command removes a specified server URL.

Syntax

```
subscriber-control filters-server-url <URL>
```

```
no subscriber-control filters-server-url
```

Parameters

<URL> – reference address, set by the string from 8 to 255 characters.

Default value

None

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# subscriber-control filters-server-url "http://192.168.1.1/files/"
```

subscriber-control peer-address

This command defines the IP address of the neighbor with which the table will be synchronized with option 82.

The use of a negative form (no) of the command removes neighbor router IP address from configuration.

Syntax

```
subscriber-control peer-address <ADDR>
no subscriber-control peer-address
```

Parameters

<ADDR> – neighbor’s IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Default value

None

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# subscriber-control peer-address 10.24.80.100
```

subscriber-control thresholds sessions-number

This command sets the threshold for the number of BRAS sessions for all user control profiles for sending snmp-trap eltexBrasSessionsNumberHigh and eltexBrasSessionsNumberHighOk.

The use of a negative form (no) of the command sets the default value.

Syntax

```
subscriber-control thresholds sessions-number { high <TH-HIGH> | low <TH-LOW> }
no subscriber-control thresholds sessions-number { high | low }
```

Parameters

<TH-HIGH> – BRAS session number threshold for sending snmp-trap eltexBrasSessionsNumberHigh;

<TH-LOW> – BRAS session number threshold for sending snmp-trap eltexBrasSessionsNumberHighOk.

Default value

On ESR-1700 <TH-HIGH> – 47000, <TH-LOW> – 46000

On ESR-1000/1200/1500/1511/3100 <TH-HIGH> – 9000, <TH-LOW> – 8500

On ESR-100/200 <TH-HIGH> – 900, <TH-LOW> – 850

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# subscriber-control thresholds sessions-number high 8000
```

subscriber-control unused-filters-remove-delay

Specify the interval after which currently unused URL lists will be removed.

The use of a negative form (no) of the command sets the default value.

Syntax

```
subscriber-control unused-filters-remove-delay <DELAY>
no subscriber-control unused-filters-remove-delay
```

Parameters

<DELAY> – time interval in seconds, takes values of [10800..86400].

Default value

10800

Required privilege level

15

Command mode

CONFIG

Example

```
esr(config)# subscriber-control unused-filters-remove-delay 40000
```

thresholds sessions-number

This command is used to set the threshold for the number of BRAS sessions to send snmp-trapeltexBrasSessionsNumberHigh and eltexBrasSessionsNumberHighOk.

The use of a negative form (no) of the command sets the default value.

Syntax

```
thresholds sessions-number { high <TH-HIGH> | low <TH-LOW> }
no thresholds sessions-number { high | low }
```

Parameters

<TH-HIGH> – BRAS session number threshold for sending snmp-trap eltexBrasSessionsNumberHigh;

<TH-LOW> – BRAS session number threshold for sending snmp-trap eltexBrasSessionsNumberHighOk.

Default value

On ESR-1700 <TH-HIGH> – 47000, <TH-LOW> – 46000

On ESR-1000/1200/1500/1511/3100 <TH-HIGH>–9000, <TH-LOW> – 8500

On ESR-20/21/100/200 <TH-HIGH> – 900, <TH-LOW> – 850

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# thresholds sessions-number high 8000
```

vrrp-group

This command defines the VRRP group, on the basis of which the state (main/standby) of the subscriber control service is defined. When the VRRP is switched to the BACKUP state, all user control sessions are reset.

The use of a negative form (no) of the command removes a VRRP identifier.

Syntax

```
vrrp-group <GRID>
```

```
no vrrp-group
```

Parameters

<GRID> – VRRP router group identifier, takes values in the range of [1..32].

Required privilege level

15

Command mode

CONFIG-SUBSCRIBER-CONTROL

Example

```
esr(config-subscriber-control)# vrrp-group 10
```

32 Tunnel Wi-Fi controller configuration

⚠ In the current firmware version, this functionality is supported only by ESR-100, ESR-200, ESR-1000, ESR-1200, ESR-1700 and ESR-3100 routers

- aaa das-profile
- aaa radius-profile
- data-tunnel configuration
- enable
- failure-count
- internal-usage-vlan
- keepalive mode
- keepalive-disable
- nas-ip-address
- peer-address
- resp-time
- retry-time
- show interfaces bridge switch-communities
- show interfaces bridge switch-communities community
- show interfaces bridge switch-communities community include
- show interfaces bridge switch-communities summary
- thresholds tunnels-number
- thresholds sub-tunnels-number
- vrrp-group
- wireless-controller

aaa das-profile

This command selects a dynamic authorization server (DAS) profile where CoA requests from the PCRF to change the service policy will be received.

The use of a negative form (no) of the command removes a specified profile of dynamic authorization servers (DAS).

Syntax

```
[no] aaa das-profile <NAME>
```

Parameters

<NAME> – DAS profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# aaa das-profile profile1
```

aaa radius-profile

This command selects the profile of the RADIUS servers to which requests will be sent to obtain the configuration of SoftGRE DATA tunnels (served by the SSID and shaping parameters).

The use of a negative form (no) of the command removes a specified RADIUS server profile.

Syntax

```
[no] aaa radius-profile <NAME>
```

Parameters

<NAME> – RADIUS server profile name, set by the string of up to 31 characters.

Required privilege level

15

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# aaa radius-profile profile1
```

data-tunnel configuration

This command sets the configuration mode for SoftGRE DATA tunnels.

The use of a negative form (no) of the command sets the default value.

Syntax

```
data-tunnel configuration { local | radius }
no data-tunnel configuration
```

Parameters

local – configuration mode in which the parameters of the SoftGRE DATA tunnels are obtained from the local configuration of the router.

radius – the mode in which the parameters of the SoftGRE DATA tunnels are requested from the RADIUS server.

Required privilege level

10

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# data-tunnel configuration radius
```

enable

This command enables Wi-Fi- controller.

The use of a negative form (no) of the command disables Wi-Fi controller.

Syntax

`[no] enable`

Parameters

The command does not contain parameters.

Default value

Process disabled.

Required privilege level

10

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# enable
```

failure-count

This command defines the number of consecutive failed ICMP requests after which, in the absence of a response from the opposite side, the tunnel is considered inoperative and is removed from the system.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] failure-count <VALUE>
```

Parameters

<COUNT> – amount of packets copies, takes values of [1..100].

Default value

5

Required privilege level

10

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# failure-count 8
```

internal-usage-vlan

This command specifies the list of vlan that will be used to form the switch-community

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] internal-usage-vlan <VLAN_List>
```

Parameters

<VLAN_List> – VLAN ID, takes the values of [2..4094].

You can specify several VLANs separated by commas ',' or you can specify the range of VLANs with '-'.

Default value

None

Required privilege level

10

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# internal-usage-vlan 1001-2037,2500-3000,3200-3700
```

keepalive mode

This command sets the keepalive mode for SoftGRE tunnels.

The use of a negative form (no) of the command sets the default value.

Syntax

```
keepalive mode { proactive | reactive }
no keepalive mode
```

Parameters

- proactive – SoftGRE tunnels check mode via icmp checking of the remote address;
- reactive – SoftGRE tunnel checking mode by receiving icmp packets with payload: "ELTEX_GRE IPv4 <tunnel device ip address>" from remote devices inside the tunnel.

Default value

proactive

Required privilege level

10

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# keepalive mode reactive
```

keepalive-disable

This command disables the exchange of ICMP messages, which are used to check the availability of a remote Wi-Fi tunnel gateway controller.

The use of a negative form (no) of the command enables ICMP messages exchange.

Syntax

```
[no] keepalive-disable
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# keepalive-disable
```

nas-ip-address

This command defines the IP address of the router that will be used as the source IP address in packets sent by RADIUS.

The use of a negative form (no) of the command removes a specified source IP address.

Syntax

```
nas-ip-address <ADDR>
no nas-ip-address
```

Parameters

<ADDR> – source IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

Required privilege level

15

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# nas-ip-address 10.100.100.2
```

peer-address

This command defines the IP address of the neighbor with which the tunnels will be reserved.

The use of a negative form (no) of the command removes neighbor router IP address from configuration.

Syntax

```
[no] peer-address <ADDR>
```

Parameters

<ADDR> – neighbor’s IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Required privilege level

10

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# peer-address 192.168.0.15
```

resp-time

This command defines the time to wait for a response, after which the ICMP request is considered rejected.

The use of a negative form (no) of the command sets the default resp-time value.

Syntax

```
[no] resp-time <TIME >
```

Parameters

<TIME> – amount of seconds, takes values of [1..30].

Default value

5

Required privilege level

10

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# resp-time 30
```

retry-time

This command sets the time between ICMP requests.

The use of a negative form (no) of the command sets the default retry-time value.

Syntax

```
[no] retry time < TIME >
```

Parameters

<TIME> – amount of seconds, takes values of [60..3600].

Required privilege level

10

Default value

60

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# retry-time 12
```

show interfaces bridge switch-communities

This command allows to view a list of switch-communities that have been formed.

Syntax

```
show interfaces bridge switch-communities [<BRIDGE-ID>]
```

Parameters

<BRIDGE-ID> – bridge ID number, specified in the form described in Section Types and naming order of router interfaces.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show interfaces bridge switch-communities
bridge 10
Location                           Community                                  Interfaces
--------------------------------   ---------------------------------------
------------------------------------------
data10                             Eltex.Novosibirsk.Novosibirskaya_oblast   softgre 2.10,
softgre 4.10, softgre 6.10,
                                   .MRF_Sibir.Local.AllClients.root           softgre 8.10

data10                             TEST.Novosibirsk.Novosibirskaya_oblast   softgre 2.310,
softgre 4.310, softgre
                                   .MRF_Sibir.Local.AllClients.root           6.310, softgre
8.310


bridge 11
Location                           Community                                  Interfaces
--------------------------------   ---------------------------------------
------------------------------------------
data11                             Eltex.Novosibirsk.Novosibirskaya_oblast   softgre 2.311,
softgre 4.311, softgre
                                   .MRF_Sibir.Local.AllClients.root           6.311, softgre
8.311
```

show interfaces bridge switch-communities community

This command allows to view a list of formed switch-communities corresponding to a particular EMS domain name.

Syntax

```
show interfaces bridge switch-communities community <EMS-DOMAIN> [<BRIDGE-ID>]
```

Parameters

<EMS-DOMAIN> – EMS domain name, WORD(1-246).

<BRIDGE-ID> – bridge ID number, specified in the form described in Section Types and naming order of router interfaces.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show interfaces bridge switch-communities community
Eltex.Novosibirsk.Novosibirskaya_oblast.MRF_Sibir.Local.AllClients.root
bridge 10
Location                           Community                                  Interfaces
--------------------------------   ---------------------------------------
------------------------------------------
data10                             Eltex.Novosibirsk.Novosibirskaya_oblast    softgre 2.10,
softgre 4.10, softgre 6.10,
                                   .MRF_Sibir.Local.AllClients.root           softgre 8.10


bridge 11
Location                           Community                                  Interfaces
--------------------------------   ---------------------------------------
------------------------------------------
data11                             Eltex.Novosibirsk.Novosibirskaya_oblast    softgre 2.311,
softgre 4.311, softgre
                                   .MRF_Sibir.Local.AllClients.root           6.311, softgre
8.311
```

show interfaces bridge switch-communities community include

This command allows to view a list of formed switch-communities corresponding to a particular pattern in the EMS domain name.

Syntax

```
show interfaces bridge switch-communities community include <WORD> [<BRIDGE-ID>]
```

Parameters

<WORD> – pattern by which the EMS domain name is searched, WORD(1-246).

<BRIDGE-ID> – bridge ID number, specified in the form described in Section Types and naming order of router interfaces.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show interfaces bridge switch-communities community include eltex
bridge 10
Location                           Community                                 Interfaces
--------------------------------   ---------------------------------------
------------------------------------------
data10                             Eltex.Novosibirsk.Novosibirskaya_oblast   softgre 2.10,
softgre 4.10, softgre 6.10,
                                   .MRF_Sibir.Local.AllClients.root          softgre 8.10


bridge 11
Location                           Community                                 Interfaces
--------------------------------   ---------------------------------------
------------------------------------------
data11                             Eltex.Novosibirsk.Novosibirskaya_oblast   softgre 2.311,
softgre 4.311, softgre
                                   .MRF_Sibir.Local.AllClients.root          6.311, softgre
8.311
```

show interfaces bridge switch-communities summary

This command allows to view number of switch-communities that have been formed.

Syntax

```
show interfaces bridge switch-communities summary [<BRIDGE-ID>]
```

Parameters

<BRIDGE-ID> – bridge ID number, specified in the form described in Section Types and naming order of router interfaces.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show interfaces bridge switch-communities summary
Bridges      Communities
----------   -----------
bridge 10    2
bridge 11    1
```

thresholds tunnels-number

This command sets the threshold for the number of SoftGRE tunnels for sending snmp-trap eltexInterfacesNumberHigh and eltexInterfacesNumberHighOk.

The use of a negative form (no) of the command sets the default value.

Syntax

```
thresholds tunnels-number { high <TH-HIGH> | low <TH-LOW> }
no thresholds tunnels-number { high | low }
```

Parameters

<TH-HIGH> – SoftGRE tunnels number threshold for sending snmp-trap eltexInterfacesNumberHigh;

<TH-LOW> – SoftGRE tunnels number threshold for sending snmp-trap eltexInterfacesNumberHighOk.

Default value

On ESR-1000 <TH-HIGH> – 900, <TH-LOW> – 850

On ESR-100 и ESR-200 <TH-HIGH> – 400, <TH-LOW> – 350

Required privilege level

10

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# thresholds tunnels-number high 1000
```

thresholds sub-tunnels-number

This command sets the threshold for the number of SoftGRE subtunnels for sending snmp-trap eltexInterfacesNumberHigh and eltexInterfacesNumberHighOk.

The use of a negative form (no) of the command sets the default value.

Syntax

```
thresholds sub-tunnels-number { high <TH-HIGH> | low <TH-LOW> }
no thresholds sub-tunnels-number { high | low }
```

Parameters

<TH-HIGH> – SoftGRE subtunnels number threshold for sending snmp-trap eltexInterfacesNumberHigh;

<TH-LOW> – SoftGRE subtunnels number threshold for sending snmp-trap eltexInterfacesNumberHighOk.

Default value

On ESR-1700, ESR-1200 <TH-HIGH> – 7000, <TH-LOW> – 7500

On ESR-3100 <TH-HIGH> – 7800, <TH-LOW> – 7600

On ESR-1000 <TH-HIGH> – 2500, <TH-LOW> – 2000

On ESR-100 и ESR-200 <TH-HIGH> – 1000, <TH-LOW> – 900

Required privilege level

10

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# thresholds sub-tunnels-number high 2000
```

vrrp-group

This command defines the VRRP group, on the basis of which the state (main/standby) of the Wi-Fi controller is defined.

The use of a negative form (no) of the command removes a VRRP identifier.

Syntax

```
vrrp-group <GRID>
no vrrp-group
```

Parameters

<GRID> – VRRP router group identifier, takes values in the range of [1..32].

Required privilege level

10

Command mode

CONFIG-WIRELESS

Example

```
esr(config-wireless)# vrrp-group 10
```

wireless-controller

Switch to Wi-Fi controller configuration mode.

The use of a negative form (no) of the command clears the configuration and disables the Wi-Fi tunnel controller.

Syntax

```
[no] wireless-controller
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# wireless-controller
```

33 VoIP configuration

⚠ This functionality is supported only by ESR-12V/12VF/14VF routers

- SIP configuration
  - Configure a SIP profile
    - sip profile
    - codec-packettime g711
    - codec-packettime g723
    - codec-packettime g729
    - conference type
    - conference uri address
    - conference uri port
    - conference uri user
    - description
    - dialplan pattern
    - echo-caceller
    - enable
    - fax protocol t38
    - fax protocol t38 redundant
    - ims call-holding
    - ims call-transfer
    - ims call-waiting
    - ims conference
    - ims hotline
    - ims type
    - payload dtmf
    - rtcp
    - rtcp extend-reports
    - rtcp keepalive
    - rtcp tx-period
    - session expiries-timer
    - session keepalive
    - session timer disable
    - silence-detect
    - sip-domain address
    - sip-domain registration enable
    - sip-domain registration retry
    - source-address
    - transfer dtmf
    - transfer flash
  - SIP PROXY configuration
    - proxy primary
    - proxy primary keepalive
    - proxy primary mode
    - proxy secondary
    - enable
    - ip address proxy-server
    - ip address registration-server
    - ip port proxy-server
    - ip port registration-server
    - registration
  - SIP common parameters configuration

        - authentication mode
        - authentication name
        - authentication password
        - sip service
    - Dial plan configuration
        - description
        - dialplan pattern
        - enable
        - pattern
    - VAS Configuration
        - prefix busy
        - prefix call-waiting
        - prefix dnd
        - prefix hotline
        - prefix no-answer
        - prefix pickup
        - prefix uncoditional
        - voice services
    - SIP ALG feature activation
        - service-voip routing
- PBX configuration
    - PBX server configuration
        - enable
        - expiry
        - external-config
        - session-timer type
        - session-timer expiries
        - session-timer min-se
        - session-timer refresher
        - profile
        - pbx
        - register-server
        - ruleset
        - user
        - transport protocol
        - transport port
        - fax
    - Configuring a subscriber on a PBX server
        - password
        - profile
    - Configuring the routing context on the PBX server
        - description
        - enable
        - pattern
        - rule
    - Configuring the registration server for PBX
        - protocol
        - authentication password
        - username
        - ip address
        - ip port
        - domain
        - profile
        - enable

- Configuring the SIP profile on a PBX server
    - type
    - client
    - from-domain
    - security level
    - nat
    - codec allow
    - qualify
    - host-address
    - sip-port
    - authentication password
    - username
    - protocol
    - ruleset
    - in-band
    - rtp-media
    - group pickup
    - group calling
- Display PBX server information
    - show pbx peers
    - show pbx registrations

SIP configuration

Configure a SIP profile

sip profile

This command creates a SIP profile and switchs to configuration mode.

The use of a negative form (no) of the command removes a SIP profile.

Syntax

```
[np] sip profile <NUM>
```

Parameters

<NUM> – SIP profile number, set in the form of a digit from 1 to 5.

Default value

None

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# sip profile 1
```

codec-packettime g711

This command sets the packetization time for the G711 codec. Packetization time – the number of milliseconds of speech in one RTP packet.

The use of a negative form (no) of the command sets the default value.

Syntax

```
codec-packettime g711 <PTIME>
no codec-packettime g711
```

Parameters

<PTIME> – packetization time in milliseconds, takes values of [10, 20, 30, 40, 50, 60].

Default value

20

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# codec-packettime g711 50
```

codec-packettime g723

This command sets the packetization time for the G723 codec. Packetization time – the number of milliseconds of speech in one RTP packet.

The use of a negative form (no) of the command sets the default value.

Syntax

```
codec-packettime g723 <PTIME>
no codec-packettime g723
```

Parameters

<PTIME> – packetization time in milliseconds, takes values of [30, 60, 90].

Default value

30

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# codec-packettime g723 60
```

codec-packettime g729

This command sets the packetization time for the G729 codec. Packetization time – the number of milliseconds of speech in one RTP packet.

The use of a negative form (no) of the command sets the default value.

Syntax

```
codec-packettime g729 <PTIME>
no codec-packettime g729
```

Parameters

<PTIME> – packetization time in milliseconds, takes values of [10, 20, 30, 40, 50, 60, 70, 80].

Default value

20

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# codec-packettime g729 70
```

conference type

This command sets the operation mode of the three-way conference.

Using the no conf_mode command sets the conference mode to default.

Syntax

```
conference type <MODE>
no conf_mode
```

Parameters

<MODE> – VPN operation mode. Two modes are available:

- local - conference is intended locally by the device after pressing 'flash+3' combination;
- remote – conference is intended on remote server. Then, after pressing 'flash+3' combination Invite message sending on server to number, pointed in the *'Conference server'* In this case conference processing by algorithm, described in RFC4579.

Default value

local

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# conference type remote
```

conference uri address

This command sets the address of the remote conference server in the SIP-URI scheme: user@address:port.

The use of a negative form (no) of the command removes the value.

Syntax

```
conference uri address { <ADDR> | <NAME> }
[no] conference uri address
```

Parameters

<ADDR> – may take the value of ipv4 address or the domain name.

IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<NAME> – remote conference server DNS name. Example of domain name – example.loc.

Default value

None

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# conference uri address sipnetwork.loc
```

conference uri port

This command sets the port of the remote conference server in the SIP-URI scheme: user@address:port.

The use of a negative form (no) of the command removes the value.

Syntax

```
conference uri port <PORT>
no conference uri port
```

Parameters

<PORT> – number of TCP/UDP port to exchange data with a remote server, takes values of [1..65535].

Default value

5060

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# conference uri port 5066
```

conference uri user

This command sets the user name of the remote conference server in the SIP-URI scheme: user@address:port.

The use of a negative form (no) of the command sets the default value.

Syntax

```
conference uri user <USER>
no conference uri user
```

Parameters

<USER> – server profile name, set by the string of up to 31 characters.

Default value

conf

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# conference uri user conference
```

description

The command is used to register on SIP server.

The use of a negative form (no) of the command removes a name of configurated SIP profile.

Syntax

```
description <DESCRIPTION>
no description
```

Parameters

<DESCRIPTION> – profile description, set by the string of up to 255 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

CONFIG-VOIP-PROFILE

Example

```
esr(config-sip-profile)# description “first sip profile”
```

dialplan pattern

This command assigns a dial plan to the current SIP profile.

The use of a negative form (no) of the command disables specified dial plan.

Syntax

```
dialplan pattern <DNAME>
no dialplan pattern
```

Parameters

<DNAME> – name of the dial plan, set by the string of up to 31 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# dialplan pattern dialplanName
```

echo-caceller

The command enables SIP proxy.

The use of a negative form (no) of the command removes the echocancelling.

Syntax

```
[no] echo-canceller
```

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# echo-canceller
```

enable

The command enables SIP proxy.

The use of a negative form (no) of the command disables SIP profile.

Syntax

`[no] enable`

Parameters

The command does not contain parameters.

Default value

SIP profile is disabled.

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# enable
```

fax protocol t38

This command allows the transition in T.38 - when an incoming re-invite to T.38 from the oncoming gateway.

The use of a negative form (no) of the command restricts transition to T.38.

Syntax

`[no] fax protocol t38`

Parameters

The command does not contain parameters.

Default value

Switch to t38 is disabled.

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# fax protocol t38
```

fax protocol t38 redundant

This command adds the redundancy into T.38 packets; value is corresponding to amount of previous packets, which is doubling in every new T.38 packet. This redundancy method is useful for case when the packets are lost in the transfer.

The use of a negative form (no) of the command removes the redundancy from T.38 packets.

Syntax

```
fax protocol t38 redundant <SIZE>
no fax protocol t38 redundant
```

Parameters

<SIZE> – the number of previous packets, which are duplicated in each new T.38 packet, can be 0, 1, 2.

Default value

0

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# fax protocol t38 redundant 2
```

ims call-holding

This command assigns the name of the Call Hold service. The command changes XML element name in Notify message body, used for transmission of commands to activate/deactivate «Call Hold» service. Example: if service name has 'call-hold' value, activation command will appear as:

<call-hold active="true"/>

and deactivation command:

<call-hold active="false"/>

The use of a negative form (no) of the command sets the default service name.

Syntax

```
ims call-holding <NAME>
```

```
no ims call-holding
```

Parameters

<NAME> – name of the *'call hold'* service, set by the string of up to 31 characters.

Default value

call-hold

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# ims call-holding hold
```

ims call-transfer

This command assigns the name of the *Call Transfer* service. The command changes XML element name in Notify message body, used for transmission of commands to activate/deactivate *'3-way-conference'* service. Example: if service name has 'call-transfer' value, activation command will appear as:

< call-transfer active="true"/>

and deactivation command:

< call-transfer active="false"/>

The use of a negative form (no) of the command sets the default service name.

Syntax

```
ims call-transfer <NAME>
```

```
no ims call-transfer
```

Parameters

<NAME> – name of the *'Call Transfer'* service, set by the string of up to 31 characters.

Default value

explicit-call-transfer

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# ims call-transfer transfer
```

ims call-waiting

This command assigns the name of the *'Call Waiting' service.* The command changes the name of XML element in Notify message body used to pass enable/disable command of *'Call Waiting'* service. Example: if service name has 'call-waiting' value, activation command will appear as:

<call-waiting active="true"/>

and deactivation command:

<call-waiting active="false"/>

The use of a negative form (no) of the command sets the default service name.

Syntax

```
ims call-waiting <NAME>
no ims call-waiting
```

Parameters

<NAME> – name of the *'Call Waiting'* service, set by the string of up to 31 characters.

Default value

call-waiting

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

CONFIG-VOIP-PROFILE

Example

```
esr(config-sip-profile)# ims call-waiting wait
```

ims conference

This command assigns *«3 Way Conference»* service name – XML element name in Notify message body, used for transmission of commands to activate/deactivate *«3 Way Conference»* service. Example: if service name has 'three-party-conference' value, activation command will appear as:

< three-party-conference active="true"/>

and deactivation command:

< three-party-conference active="false"/>

The use of a negative form (no) of the command sets the default service name.

Syntax

`ims conference <NAME>`

`no ims conference`

Parameters

<NAME> – name of the *'3 Way Conference'* service, set by the string of up to 31 characters.

Default value

three-party-conference

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# ims conference conf
```

ims hotline

This command assigns *'Hotline'* service name – XML element name in Notify message body, used for transmission of commands to activate *'Hotline'* service. Activation command sending Hotline phone number and call timeout. Example: if service name has 'hot-line-service' value and it is needed to perform a call to number 30001 after 6 seconds after onhook, activation command will appear as:

<hot-line-service>

<addr>30001</addr>

<timeout>6</timeout>

</hot-line-service>

The use of a negative form (no) of the command sets the default service name.

Syntax

```
ims hotline <NAME>
```

```
no ims hotline
```

Parameters

<NAME> – name of the *'Hotline'* service, set by the string of up to 31 characters.

Default value

hot-line-service

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# ims hotline hot
```

ims type

The command assigns operation mode of VAS with IMS.

The use of a negative form (no) of the command disables the limitations.

Syntax

```
ims type <MODE>
```

```
no ims type
```

Parameters

<MODE> – Suppl with IMS operation mode, may take the following values:

- implicit - allows management of certain types of services by the IMS (*IP Multimedia Subsystem*) server. In this case enabling the *'3-way Conference'* (works only by the algorithm RFC4579), *'Call Hold', 'Call Waiting', 'Hot line'* (regardless of whether they are enabled or not in the configuration) services is performed remotely by the IMS server by sending Notify messages, the body of which sends commands to activate/deactivate services in XCAP format (in fact, XML, RFC4825). In this subscription option, gateway will send SUBSCRIBE requests after subscriber registration, and upon successful subscription, will process NOTIFY requests received from IMS, which are used for service management. *'Call Hold'* service name - XML element name in Notify message body, used for transmission of commands to activate/deactivate *'Call Hold'* For example, if the service name is «call-hold», the activation command would look like this:
  <call-hold active="true"/>

deactivation command:
<call-hold active="false"/>

- explicit - allows management of certain types of services by the IMS (*IP Multimedia Subsystem*) server. In this case enabling the *'3-way Conference'* (works only by the algorithm RFC4579), *'Call Hold', 'Call Waiting', 'Hot line'* (regardless of whether they are enabled or not in the configuration) services is performed remotely by the IMS server by sending Notify messages, the body of which sends commands to activate/deactivate services in XCAP format (in fact, XML, RFC4825). In this subscription option, gateway will not send SUBSCRIBE requests after subscriber registration, and will only process NOTIFY requests received from IMS, which are used for service management.

Default value

VAS with IMS is disabled.

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# ims type implicit
```

payload dtmf

This command assigns the type of load for transmitting packets over RFC2833.

The use of a negative form (no) of the command sets the default value.

Syntax

```
payload dtmf <TYPE>
no payload dtmf
```

Parameters

<TYPE> – type of load for transmitting packets over RFC2833, the allowed values for use are [96..127].

Default value

96

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# payload dtmf 101
```

rtcp

This command enables the RTCP protocol to control the conversational channel.

The use of a negative form (no) of the command restricts the use of RTCP.

Syntax

`[no] rtcp`

Parameters

The command does not contain parameters.

Default value

Deactivating/disabling the RTCP protocol.

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# rtcp
```

rtcp extend-reports

This command enables the sending of RTCP Extended Reports packets in accordance with RFC 3611.

The use of a negative form (no) of the command restricts the use of RTCP Extended Reports.

Syntax

`[no] rtcp extend-reports`

Parameters

The command does not contain parameters.

Default value

RTCP Extended Reports packets sending disabled.

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# rtcp extend-reports
```

rtcp keepalive

This command sets the receive period of the RTCP message, which is measured in units of the transmission interval. If at the end of the reception period no RTCP packet is received from the opposite side, ESR-12V/12VF/14VF disconnects.

The use of a negative form (no) of the command sets the default value.

Syntax

```
rtcp keepalive <TIME>
```

```
no rtcp timer
```

Parameters

<TIME> – RTCP message reception period in seconds, allowed to use the value [1..30].

Default value

5

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# rtcp keepalive 25
```

rtcp tx-period

This command sets the interval for transmitting RTCP packets.

The use of a negative form (no) of the command sets the default value.

Syntax

```
rtcp tx-period <TIME>
no rtcp period
```

Parameters

<TIME> – RTCP packet transmission interval in seconds, the allowed values for use are [1..60].

Default value

5

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# rtcp tx-period 40
```

session expiries-timer

This command sets the minimum interval for checking the operability of a connection according to RFC2028 (from 90 to 1800 s, by default 120 s).

The use of a negative form (no) of the command sets the default value.

Syntax

```
session expiries-timer <TIME>
no session expiries-timer
```

Parameters

<TIME> – the minimum interval of checking the operability of the connection in seconds, can take values [90..1800].

Default value

120

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# session expiries-timer 180
```

session keepalive

This coomand sets the period of time in seconds that should pass before the forced session termination, if the session is not renewed in time.

The use of a negative form (no) of the command sets the default value.

Syntax

```
session keepalive <TIME>
```

```
no session keepalive
```

Parameters

<TIME> – the minimum interval of checking the operability of the connection, can take values [90..80000].

Default value

1800

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# session keepalive 900
```

session timer disable

This command disables support for the timer extension (RFC 4028).

The use of a negative form (no) of the command allows the use of the update session according to RFC4028.

Syntax

```
[no] session timer disable
```

Parameters

The command does not contain parameters.

Default value

Session control is disabled.

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# session timer disable
```

silence-detect

This command enables the use of the silence detector.

The use of a negative form (no) of the command restricts the use of silence detector.

Syntax

`[no] silence-detect`

Parameters

The command does not contain parameters.

Default value

Disabled.

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# silence-detect
```

sip-domain address

This command specifies the SIP domain in which the device is located.

The use of a negative form (no) of the command removes the value of SIP domain.

Syntax

```
sip-domain address { <ADDRESS> | <NAME> }
no sip-domain address
```

Parameters

<ADDRESS> – SIP domain in which the device is located, set by ipv4 address or domain name.

<NAME> – SIP domain DNS name. Example of domain name – example.loc.

Default value

None

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# sip-domain address sip.network.com
```

sip-domain registration enable

This command enables the use of the SIP Domain during registration (the SIP domain will be inserted into the Request-Line Register requests).

The use of a negative form (no) of the command cancels the use of SIP Domain when registering.

Syntax

```
[no] sip-domain registration enable
```

Parameters

The command does not contain parameters.

Default value

Do not use of SIP domain when registering.

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# sip-domain registration enable
```

sip-domain registration retry

This command specifies the period of time during which registration of the subscriber port on the SIP server is valid. Average, port registration renewal is carried out after 2/3 of specified period.

The use of a negative form (no) of the command sets the default value.

Syntax

```
sip-domain registration retry <INTERVAL>
no sip-domain registration retry
```

Parameters

<INTERVAL> – period of time in seconds during which the registration of the subscriber port on the SIP server is valid. May take values in the range of [30..10000].

Default value

1800

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# sip-domain registration retry 900
```

source-address

This command specifies a *public address* – this parameter is used as external address of the device when operating behind NAT (behind gateway). The address of external (WAN) gateway interface (NAT), behind that ESR-12V/12VF/14VF is installed, is specified as the public address.

The use of a negative form (no) of the command removes the value.

Syntax

```
source-address <IPv4 address>
no source-address
```

Parameters

<IPv4 address> – address of the external (WAN) interface of the gateway (NAT) behind which the ESR-12V/12VF/14VF is installed.

Default value

None

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# source-address 232.104.45.12
```

transfer dtmf

The command is used to specify the method of DTMF signal transmission.

The use of a negative form (no) of the command sets the default value.

Syntax

`transfer dtmf <MODE>`

`no transfer dtmf`

Parameters

<MODE> – DTMF signals transmission mode, may take the following values:

- info – передача сообщений по протоколу SIP в запросах INFO;
- inband – внутриполосная передача;
- rfc-2833 – согласно рекомендации RFC2833 в качестве выделенной нагрузки в речевых пакетах RTP.

Default value

rfc-2833

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# transfer dtmf inband
```

transfer flash

The command is used to specify the method of Flash signal transmission.

The use of a negative form (no) of the command sets the default value.

Syntax

```
transfer flash <MODE>
```

```
no transfer flash
```

Parameters

<MODE> – Flash signals transmission mode, may take the following values:

- sscc – message transmission to a cooperating party via SIP in INFO requests. *Flash* event is sent in *Application/sscc* extension as *'event flashhook'*.
- dtmf – message transmission to a cooperating party via SIP in INFO requests. *Flash* event is sent in *Application/dtmf-relay* as *'signal=hf'*.
- hook-flash – message transmission to a cooperating party via SIP in INFO requests. *Flash* event is sent in *Application/Hook Flash* extension as *'signal=hf'*.
- broadsoft – message transmission to a cooperating party via SIP in INFO requests. *Flash* event is transmitted in *Application/Broadsoft* extension as *event flashhook*.

Default value

hook-flash

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# transfer flash dtmf
```

SIP PROXY configuration

proxy primary

The command is used to switch to the primary SIP proxy configuration mode.

The use of a negative form (no) of the command removes all settings of primary SIP proxy.

Syntax

```
[no] proxy primary
```

Parameters

The command does not contain parameters.

Default value

None

Required privilege level

15

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# proxy primary
```

proxy primary keepalive

The command specifies the primary server supervision period.

The use of a negative form (no) of the command sets the default value.

Syntax

```
proxy primary keepalive <PERIOD>
no proxy primary keepalive
```

Parameters

<PERIOD> – periodic messages sending interval in seconds to check the availability of a primary SIP server, may take the values of [10..3600].

Default value

30

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# proxy primary keepalive 60
```

proxy primary mode

The command specifies SIP proxy usage mode.

The use of a negative form (no) of the command sets the default value.

Syntax

```
proxy primary mode <MODE>
no proxy primary mode
```

Parameters

<MODE> – SIP proxy usage mode, may take the following values:

- parking – SIP proxy redundancy mode without control of primary SIP proxy;
- homing – SIP proxy redundancy mode with control of primary SIP proxy.

Default value

homing

Required privilege level

15

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# proxy primary mode parking
```

proxy secondary

The command is used to switch to the redundant SIP proxy configuration mode.

The use of a negative form (no) of the command removes all settings of redundant SIP proxy.

Syntax

```
proxy secondary <NUM>
no proxy secondary
```

Parameters

<NUM> – redundant SIP proxy number, takes values of [1..4].

Default value

None

Required privilege level

10

Command mode

CONFIG-SIP-PROFILE

Example

```
esr(config-sip-profile)# proxy secondary 2
```

enable

The command enables SIP proxy.

The use of a negative form (no) of the command disables SIP proxy.

Syntax

`[no] enable`

Parameters

The command does not contain parameters.

Default value

SIP proxy is disabled.

Required privilege level

10

Command mode

CONFIG-VOIP-SIP-PROXY

Example

```
esr(config-voip-sip-proxy)# enable
```

ip address proxy-server

The command specifies the IP address of a proxy server.

The use of a negative form (no) of the command removes the value.

Syntax

```
ip address proxy-server { <ADDR> | <NAME> }
no ip address proxy-server
```

Parameters

<ADDR> – proxy server address, set by ipv4 address or the domain name. May take the value of ipv4 address or the domain name.

IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<NAME> – proxy server DNS name. Example of domain name – example.loc.

Default value

None

Required privilege level

10

Command mode

CONFIG-VOIP-SIP-PROXY

Example

```
esr(config-voip-sip-proxy)# ip address proxy-server 10.245.56.12
```

ip address registration-server

The command specifies the log server address.

The use of a negative form (no) of the command removes the value.

Syntax

```
ip address registration-server { <ADDR> | <NAME> }
no ip address registration-server
```

Parameters

<ADDR> – log server address, set by ipv4 address or the domain name. May take the value of ipv4 address or the domain name.

IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

<NAME> – registration server DNS name. Example of domain name – example.loc.

Default value

None

Required privilege level

10

Command mode

CONFIG-VOIP-SIP-PROXY

Example

```
esr(config-voip-sip-proxy)# ip address registration-server 10.245.56.12
```

ip port proxy-server

The command specifies the number of proxy server UDP port.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip port proxy-server <PORT>
no ip port proxy-server
```

Parameters

<PORT> – number of proxy server UDP port, takes values of [1..65535].

Default value

5060

Required privilege level

10

Command mode

CONFIG-VOIP-SIP-PROXY

Example

```
esr(config-voip-sip-proxy)# ip port proxy-server 5080
```

ip port registration-server

The command specifies the number of log server UDP port.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ip port registration-server <PORT>
```

```
no ip port registration-server
```

Parameters

<PORT> – number of registration server UDP port, takes values of [1..65535].

Default value

5060

Required privilege level

10

Command mode

CONFIG-VOIP-SIP-PROXY

Example

```
esr(config-voip-sip-proxy)# ip port registration-server 5080
```

registration

The command is used to register on SIP server.

The use of a negative form (no) of the command disables the registration on SIP server.

Syntax

```
[no] registration
```

Parameters

The command does not contain parameters.

Default value

Registration is not used.

Required privilege level

10

Command mode

CONFIG-VOIP-SIP-PROXY

Example

```
esr(config-voip-sip-proxy)# registration
```

SIP common parameters configuration

authentication mode

The command is used to assign authentication profile.

The use of a negative form (no) of the command sets the default value.

Syntax

```
authentication mode <MODE>
no authentication mode
```

Parameters

<MODE> – authentication profile, may take the following values:

- global – one user name/password pair is used for all ports;
- user-defined – each port has its own user name/password value.

Default value

user-defined

Required privilege level

15

Command mode

CONFIG-SIP

Example

```
esr(config-sip)# authentication mode global
```

authentication name

This command assigns a login in case when authentication mode has the 'global' value.

The use of a negative form (no) of the command removes the value.

Syntax

```
authentication name <LOGIN>
no authentication name
```

Parameters

<LOGIN> – login for authentication, set by the string of up to 31 characters.

Default value

None

Required privilege level

15

Command mode

CONFIG-SIP

Example

```
esr(config-sip)# authentication name esr12V
```

authentication password

This command assigns a password in case when authentication mode has 'global' value.

The use of a negative form (no) of the command removes the value.

Syntax

```
authentication password { <PASS> | encrypted <ENCRYPTED-PASS>}
no authentication password
```

Parameters

<PASS> – authentication password, set by the string of up to 16 characters.

<ENCRYPTED-PASS> – hash password via sha512 algorithm, set by the string of [2..32] characters.

Default value

None

Required privilege level

15

Command mode

CONFIG-SIP

Example

```
esr(config-sip)# authentication password superpass
```

sip service

The command allows to switch to the SIP common settings configuration mode.

The use of a negative form (no) of the command returns the default common settings.

Syntax

`[no] sip services`

Parameters

The command does not contain parameters.

Default value

None

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# sip services
esr(config-sip)#
```

Dial plan configuration

description

The command is used to change a dial plan description.

The use of a negative form (no) of the command disables a dial plan description.

Syntax

`description <DESCRIPTION>`

`no description`

Parameters

<DESCRIPTION> – dial plan description, set by the string of up to 255 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG-DIAL-RULESET

Example

```
esr(config-dial-ruleset)# description first
```

dialplan pattern

This command creates a dial plan and to switch to the edit mode.

The use of a negative form (no) of the command removes the dial plan.

Syntax

```
dialplan pattern <NAME>
no dialplan pattern
```

Parameters

<NAME> – name of the dial plan, set by the string of up to 31 characters.

Default value

None

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# dialplan pattern newplan
```

enable

The command enables a dial plan being configured.

The use of a negative form (no) of the command disables a dial plan being configured.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Default value

The dial plan is not enabled.

Required privilege level

10

Command mode

CONFIG-DIAL-RULESET

Example

```
esr(config-dial-ruleset)# enable
```

pattern

This command adds the dial plan rules.

The use of a negative form (no) of the command sets the default value.

Syntax

```
pattern <REGEXP>
no pattern
```

Parameters

<REGEXP> - regular expression specifying the dial plan. Set by the string of up to 1024 characters. The rules for creating regular expressions are described in section Dial plan configuration example of user manual.

Default value

"S5, L5 ([xABCD*#].S)"

Required privilege level

10

Command mode

CONFIG-DIAL-RULESET

Example

```
esr(config-dial-ruleset)# pattern "S5, L5 (41xx@{local} | 7383xxxxxxx)"
```

VAS Configuration

prefix busy

This command specifies the code, dialed from a telephone, for activation, deactivation and *'Call forwarding on busy'* services configuration.

The use of a negative form (no) of the command removes the value.

Syntax

```
prefix busy <CODE>
```

```
no prefix busy
```

Parameters

<CODE> – *'Call forwarding on busy'* service configuration code dialed from a telephone, can take values of [10..99].

Default value

None

Required privilege level

10

Command mode

CONFIG-VOIP-SERVICES

Example

```
esr(config-voip-services)# prefix busy 99
```

prefix call-waiting

This command specifies the code, dialed from a telephone, for activation, deactivation and *'Call waiting'* services configuration.

The use of a negative form (no) of the command removes the value.

Syntax

```
prefix call-waiting <CODE>
```

```
no prefix call-waiting
```

Parameters

<CODE> – code dialed from the telephone to control the 'Call Waiting' service can be [10..99].

Default value

None

Required privilege level

10

Command mode

CONFIG-VOIP-SERVICES

Example

```
esr(config-voip-services)# prefix call-waiting 89
```

prefix dnd

The command is used to specify the code, dialed from a telephone, for activation, deactivation and *'Do not disturb'* service configuration.

The use of a negative form (no) of the command removes the value.

Syntax

```
prefix dnd <CODE>
no prefix dnd
```

Parameters

<CODE> – *'Do not disturb'* service configuration code dialed from a telephone, can take values of [10..99].

Default value

None

Required privilege level

10

Command mode

CONFIG-VOIP-SERVICES

Example

```
esr(config-voip-services)# prefix dnd 79
```

prefix hotline

The command is used to specify the code, dialed from a telephone, for activation, deactivation and *'Hot/Warm line'* service configuration.

The use of a negative form (no) of the command removes the value.

Syntax

```
prefix hotline <CODE>
no prefix hotline
```

Parameters

<CODE> – *'Hot/Warm line'* service configuration code dialed from a telephone, can take values of [10..99].

Default value

None

Required privilege level

10

Command mode

CONFIG-VOIP-SERVICES

Example

```
esr(config-voip-services)# prefix hotline 69
```

prefix no-answer

This command specifies the code, dialed from a telephone, for activation, deactivation and *'Call forwarding - No Answer'* services configuration.

The use of a negative form (no) of the command removes the value.

Syntax

```
prefix no-answer <CODE>
```

```
no prefix no-answer
```

Parameters

<CODE> – 'Call forwarding - No Answer' service configuration code dialed from a telephone, can take values of [10..99].

Default value

None

Required privilege level

10

Command mode

CONFIG-VOIP-SERVICES

Example

```
esr(config-voip-services)# prefix no-answer 59
```

prefix pickup

The command is used to specify the code, dialed from a telephone, for activation, deactivation and *'Call pickup'* services configuration.

The use of a negative form (no) of the command removes the value.

Syntax

```
prefix pickup <CODE>
no prefix pickup
```

Parameters

<CODE> – *'Call pickup'* service configuration code dialed from a telephone, can take values of [10..99].

Default value

None

Required privilege level

10

Command mode

CONFIG-VOIP-SERVICES

Example

```
esr(config-voip-services)# prefix pickup 49
```

prefix uncoditional

This command specifies the code, dialed from a telephone, for activation, deactivation and *'Call Forward Unconditional'* services configuration.

The use of a negative form (no) of the command removes the value.

Syntax

```
prefix uncoditional <CODE>
no prefix uncoditional
```

Parameters

<CODE> – *'Unconditional forwarding'* service configuration code dialed from a telephone, can take values of [10..99].

Default value

None

Required privilege level

10

Command mode

CONFIG-VOIP-SERVICES

Example

```
esr(config-voip-services)# prefix uncoditional 39
```

voice services

The command is used to switch to the VAS configuration mode.

The use of a negative form (no) of the command returns the default configuration.

Syntax

```
[no] voice services
```

Parameters

The command does not contain parameters.

Default value

None

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# voice services
```

SIP ALG feature activation

service-voip routing

This command activates the feature of routing VoIP packets when the PBX server is disconnected.

The use of a negative form (no) of the command disables the feature.

Syntax

```
[no] service-voip routing
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-E1

CONFIG-IF-GI

CONFIG-SUBIF

CONFIG-QINQ-IF

CONFIG-MULTILINK

CONFIG-PORT-CHANNEL

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-VTI

CONFIG-BRIDGE

Example

```
esr(config-if-gi)# service-voip routing
```

PBX configuration

PBX server configuration

enable

This command activates the embedded PBX server.

The use of a negative form (no) of the command disables the embedded PBX server and erases settings.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-PBX

Example

```
esr (config-pbx)# enable
```

expiry

This command controls the validity period of registration.

The use of a negative form (no) of the command sets the default value.

Syntax

`[no] expiry [min|max] <EMIN>`

Parameters

min – minimum registration duration;

max – maximum registration duration;

<EMIN> – may take values of [60…64800] in seconds.

Default value

min – 60;

max – 3600.

Required privilege level

10

Command mode

CONFIG-PBX

Example

```
esr (config-pbx)# expiry min 90
```

external-config

This command is used to activate the use of an alternate configuration.

The use of the negative form of the command (no) deactivates this function.

Syntax

```
[no] external-config
```

Parameters

The command does not contain parameters.

Default value

Deactivated

Required privilege level

15

Command mode

CONFIG-PBX

Example

```
esr(config-pbx)# external-config
```

session-timer type

This command configures session timer start parameters.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] session-timer type <TTIME>
```

Parameters

<TTIME> – may take one of the values: [originate, accept, refuse];

- originate – always start session timer;
- accept – start the timer only at the request of the other side;
- refuse – do not start session timer.

Default value

refuse

Required privilege level

10

Command mode

CONFIG-PBX

Example

```
esr (config-pbx)# session-timer type accept
```

session-timer expiries

This command sets the maximum interval for updating a session.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] session-timer expiries <ETIME>
```

Parameters

<ETIME> – maximum interval for updating a session in seconds, may take values [90-80000].

Default value

1800.

Required privilege level

10

Command mode

CONFIG-PBX

Example

```
esr (config-pbx)# session-timer expiries 600
```

session-timer min-se

This command sets the minimum interval for updating a session.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] session-timer min-se <MTIME>
```

Parameters

<MTIME> – minimum interval for updating a session in seconds, may take values [90-1800].

Default value

90 seconds.

Required privilege level

10

Command mode

CONFIG-PBX

Example

```
esr (config-pbx)# session-timer min-se 100
```

session-timer refresher

This command sets the initiator for updating a session.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] session-timer refresher <RTIME>
```

Parameters

<RTIME> – session refresh initiator, may take value: (uac|uas);

uac – caller initiates a session update when possible;

uas – callee initiates a session update when possible.

Default value

uas

Required privilege level

10

Command mode

CONFIG-PBX

Example

```
esr (config-pbx)# session-timer refresher uas
```

profile

This command is used to configure the SIP profile in PBX.

The use of a negative form (no) of the command removes the profile.

Syntax

`[no] profile <PNAME>`

Parameters

<PNAME> – profile name, specified by a word from 1 to 31 characters long.

Required privilege level

10

Command mode

CONFIG-PBX

Example

```
esr (config-pbx)# profile sip_phones
```

pbx

This command is used to configure the internal pbx server.

The use of a negative form (no) of the command disables the use of internal pbx server.

Syntax

`[no] pbx`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG

Example

```
esr(config)# pbx
```

register-server

This command configures registration server data in PBX.

The use of a negative form (no) of the command removes registration server information.

Syntax

```
[no] register-server <REGNAME>
```

Parameters

<REGNAME> – Registration server name, specified by a word from 1 to 31 characters long.

Required privilege level

10

Command mode

CONFIG-PBX

Example

```
esr (config-pbx)# register-server ssw
```

ruleset

This command configures the routing context in PBX.

The use of a negative form (no) of the command removes routing context.

Syntax

```
[no] ruleset <RULNAME>
```

Parameters

<RULENAME> – routing context name, specified by a word from 1 to 31 characters long.

Required privilege level

10

Command mode

CONFIG-PBX

Example

```
esr (config-pbx)# ruleset local
```

user

This command configures the subscribers in PBX.

The use of a negative form (no) of the command removes the subscriber.

Syntax

```
[no] user <UNAME>
```

Parameters

<UNAME> – subscriber name or number, specified by string from 1 to 31 characters long.

Required privilege level

15

Command mode

CONFIG-PBX

Example

```
esr(config-pbx)# user 1482
```

transport protocol

This command selects the transport protocol used by PBX.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] transport protocol <PROTO>
```

Parameters

<PROTO> – used transport protocol, may take the value [tcp | udp | both].

Default value

udp

Required privilege level

10

Command mode

CONFIG-PBX

Example

```
esr (config-pbx)# transport protocol udp
```

transport port

This command configures the transport port used by PBX.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] transport port <PORT>
```

Parameters

<PORT> – port number, takes values of [1..65535].

Default value

5060

Required privilege level

10

Command mode

CONFIG-PBX

Example

```
esr (config-pbx)# transport port 5060
```

fax

This command configures the fax parameters in PBX.

The use of a negative form (no) of the command removes a current value.

Syntax

```
[no] fax protocol t38 error-correction <FTYPE> datagram-size <SIZE>
```

Parameters

<FTYPE> – the choice of error correction method, may take the values [fec | redundancy];

<SIZE> – the choice of the datagram size in bytes, may take the value [272 ... 512].

Required privilege level

10

Command mode

CONFIG-PBX

Example

```
esr (config-pbx)# fax protocol t38 error-correction fec datagram-size 272
```

Configuring a subscriber on a PBX server

password

This command configures PBX subscriber authentication parameters.

The use of a negative form (no) of the command removes the value.

Syntax

```
password { <PASS> | encrypted <ENCRYPTED-PASS>}
no password
```

Parameters

<PASS> – authentication password, set by the string of up to 16 characters.

<ENCRYPTED-PASS> – hash password via sha512 algorithm, set by the string of [2..32] characters.

Required privilege level

15

Command mode

CONFIG-PBX-USER

Example

```
esr (config-pbx-user)# password 1234
```

profile

This command is used to assign the SIP profile to PBX subscriber.

The use of a negative form (no) of the command removes a current configuration.

Syntax

```
[no] profile <PNAME>
```

Parameters

<PNAME> – SIP profile name, specified by a word from 1 to 31 characters long.

Required privilege level

10

Command mode

CONFIG-PBX-USER

Example

```
esr (config-pbx-user)# profile sip_phones
```

Configuring the routing context on the PBX server

description

This command changes the description of the PBX routing context or rule.

The use of a negative form (no) of the command removes description.

Syntax

`[no] description <DESCR>`

Parameters

<DESCR>– description of the routing context or rule, specified by a string from 1 to 255 characters.

Required privilege level

10

Command mode

CONFIG-PBX-RULESET

CONFIG-PBX-RULE

Example

```
esr (config-pbx-ruleset)# description local_calls
```

enable

This command activates the current routing rule.

The use of a negative form (no) of the command disables the rule.

Syntax

`[no] enable`

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-PBX-RULE

Example

```
esr (config-pbx-rule)# enable
```

pattern

This command creates a pattern in a routing rule.

The use of a negative form (no) of the command removes the pattern.

Syntax

`[no] pattern <PSTR>`

Parameters

<PSTR> – routing rule pattern, specified by a string from 1 to 256 characters.

Default value

(null)

Required privilege level

10

Command mode

CONFIG-PBX-RULE

Example

```
esr (config-pbx-rule)# pattern _XXX,1,Dial(SIP/${EXTEN})
```

rule

This command configures the routing rule in the current context.

The use of a negative form (no) of the command removes the rule.

Syntax

`[no] rule <RIDX>`

Parameters

<RIDX> – the index of the rule in the context of routing, specified by a number from 1 to 1000.

Required privilege level

10

Command mode

CONFIG-PBX-RULESET

Example

```
esr (config-pbx-ruleset)# rule 1
```

Configuring the registration server for PBX

protocol

This command is used to specify the protocol for communication with the registration server.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] protocol <PROTO>
```

Parameters

<PROTO> – the protocol used to communicate with the registration server, may be {tcp | udp}.

Default value

udp

Required privilege level

10

Command mode

CONFIG-PBX-REG-SERVER

Example

```
esr (config-pbx-reg-server)# protocol udp
```

authentication password

This command configures authentication on the registration server.

The use of a negative form (no) of the command removes the value.

Syntax

```
authentication password { <PASS> | encrypted <ENCRYPTED-PASS>}
no authentication password
```

Parameters

<PASS> – authentication password, set by the string of up to 16 characters.

<ENCRYPTED-PASS> – hash password via sha512 algorithm, set by the string of [2..32] characters.

Required privilege level

10

Command mode

CONFIG-PBX-REG-SERVER

Example

```
esr (config-pbx-reg-server)# authentication password 1234
```

username

This command specifies the username on the upstream registration server.

The use of a negative form (no) of the command removes a current value.

Syntax

```
[no] username <UNAME>
```

Parameters

<UNAME> – authentication username on the registration server.

Required privilege level

10

Command mode

CONFIG-PBX-REG-SERVER

Example

```
esr (config-pbx-reg-server)# username esr
```

ip address

The command specifies the log server address.

The use of a negative form (no) of the command removes a current value.

Syntax

`[no] ip address <ADDR>`

Parameters

<ADDR> – the registration server address, specified in A.B.C.D format, where it takes the value [0..255], or with a domain name from 1 to 31 characters long.

Required privilege level

10

Command mode

CONFIG-PBX-REG-SERVER

Example

```
esr (config-pbx-reg-server)# ip address eltex-co.ru
```

ip port

The command specifies the log server port.

The use of a negative form (no) of the command sets the default value.

Syntax

`[no] ip port <PORT>`

Parameters

<PORT> – destination port on the registration server, may take the value [1 ... 65535].

Default value

5060

Required privilege level

10

Command mode

CONFIG-PBX-REG-SERVER

Example

```
esr (config-pbx-reg-server)# ip port 5060
```

domain

This command specifies the SIP domain on the registration server.

The use of a negative form (no) of the command removes a current value.

Syntax

`[no] domain <DSTR>`

Parameters

<DSTR> – SIP domain on the registration server, specified by a word from 1 to 31 characters long.

Required privilege level

10

Command mode

CONFIG-PBX-REG-SERVER

Example

```
esr(config-pbx-reg-server)# ssw3.11
```

profile

This command specifies the SIP profile used when accessing the registration server.

The use of a negative form (no) of the command removes a current value.

Syntax

`[no] profile <PNAME>`

Parameters

<PNAME> – SIP profile assigned to the registration server, specified by a word from 1 to 31 characters long.

Required privilege level

10

Command mode

CONFIG-PBX-REG-SERVER

Example

```
esr(config-pbx-reg-server)# profile sip_phones
```

enable

This command activates registration on a higher server.

The use of a negative form (no) of the command disables a current configuration.

Syntax

```
[no] enable
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-PBX-REG-SERVER

Example

```
esr (config-pbx-reg-server)# enable
```

Configuring the SIP profile on a PBX server

type

This command specifies the type created on the PBX SIP profile.

The use of a negative form (no) of the command removes a current value.

Syntax

```
[no] type <TTYPE>
```

Parameters

<TTYPE> – type of the created SIP profile, may take the values {local | remote};

- local – profile is used to register on the local server;
- remote – profile is used to register on the remote server.

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# type local
```

client

This command specifies the type of clients using the SIP profile.

The use of a negative form (no) of the command sets the default value.

Syntax

`[no] client <CTYPE>`

Parameters

<CTYPE> – type of the user connected to the profile can take one of the values {peer | friend | user};

- peer – incoming and outgoing calls are allowed without authorisation;
- user – only incoming calls are allowed;
- friend – combines peer and user profile types.

Default value

peer

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# client friend
```

from-domain

This command assigns a SIP profile to a domain.

The use of a negative form (no) of the command removes a current value.

Syntax

`[no] from-domain <FDSTR>`

Parameters

<FDSTR> – domain name, specified by a word from 1 to 31 characters long.

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# from-domain sip.domain
```

security level

This command defines how to handle connections with peers.

The use of a negative form (no) of the command removes a current value.

Syntax

```
[no] security level <STYPE>
```

Parameters

<STYPE> – takes the value {port | invite | invire-port | address | support};

- port – allow checking peer compliance by ip-address without checking the port number;
- invite – do not require authentication of incoming INVITE messages;
- invite-port – combines port and invite;
- address – authentication only by IP authentication;
- support – allow calls without re-authentication.

Default value

None

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# security level address
```

nat

This command defines the interaction policy with NAT.

The use of a negative form (no) of the command removes a current value.

Syntax

`[no] nat <NTYPE>`

Parameters

<NTYPE> – policy of interaction with NAT, may take values {comedia | force-port | both};

- comedia – send media stream to PBX port, regardless of SDP instructions;
- force-port – use rport even if it is not present;
- both – combines comedia and force-port.

Default value

None

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# nat comedia
```

codec allow

This command specifies the codecs used by the SIP profile.

The use of a negative form (no) of the command removes a current value.

Syntax

`[no] codec allow <ATYPE>`

Parameters

<ATYPE> – type of codec allowed, may take values [G711A | G711U | G722 | G726].

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# codec allow g722
```

qualify

This command configures client availability check.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] qualify <QTYPE>
```

Parameters

<QTYPE> – period of checking the availability of the client in milliseconds can take the value [100 ... 100000].

Default value

1000

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# qualify 100000
```

host-address

This command specifies the addresses of subscribers associated with this profile.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] host-address <ADDR>
```

Parameters

<ADDR> – subscriber address, specified in A.B.C.D format, where it takes the value [0..255], or with a domain name from 1 to 31 characters long.

Default value

dynamic

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# host-address 192.168.116.183
```

sip-port

This command specifies the port of subscribers associated with this profile.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] sip-port <PORT>
```

Parameters

<PORT> – subscriber port, set by number in the range of [1..65535].

Default value

5060

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# sip-port 5062
```

authentication password

This command configures authentication of users with this profile.

The use of a negative form (no) of the command removes a current value.

Syntax

```
authentication password { <PASS> | encrypted <ENCRYPTED-PASS>}
no authentication password
```

Parameters

<PASS> – authentication password, set by the string of up to 16 characters.

<ENCRYPTED-PASS> – hash password via sha512 algorithm, set by the string of [2..32] characters.

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr(config-pbx-profile)# authentication password 1234
```

username

This command configures username for subscribers with this profile.

The use of a negative form (no) of the command removes a current value.

Syntax

```
[no] username <UNAME>
```

Parameters

<UNAME> – authentication name for users, specified by a word from 1 to 31 characters.

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr(config-pbx-profile)# username user
```

protocol

This command specifies the transport protocol used by subscribers with this profile.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] protocol <PTYPE>
```

Parameters

<PTYPE> – used transport protocol, may take the value [tcp | udp | both].

Default value

udp

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# protocol udp
```

ruleset

This command specifies the routing context to which this profile is bound.

The use of a negative form (no) of the command removes a current value.

Syntax

```
[no] ruleset <RNAME>
```

Parameters

<RNAME> – routing context associated with this profile. Specified by a word from 1 to 31 characters.

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# ruleset local
```

in-band

This command configures the in-band signaling.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] in-band <ITYPE>
```

Parameters

<ITYPE> – may take values { always | not | never };

- Always – always use in-band signaling;
- Not – do not use in-band signaling;
- Never – never use in-band signaling.

Default value

never

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# in-band not
```

rtp-media

This command enables the direction of RTP traffic directly between peers.

The use of a negative form (no) of the command removes a current value.

Syntax

```
[no] rtp-media
```

Parameters

The command does not contain parameters.

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# rtp-media
```

group pickup

This command assigns the SIP profile to the capture group.

The use of a negative form (no) of the command sets the default value.

Syntax

```
group pickup <GTYPE>
no group pick-up
```

Parameters

<GTYPE> – pickup group, specified by the number from 1 to 31.

Default value

255

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# group pickup 16
```

group calling

This command assigns the SIP profile to the call group.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] group calling <GTYPE>
```

Parameters

<GTYPE> – call group, specified by the number from 1 to 31.

Default value

255

Required privilege level

10

Command mode

CONFIG-PBX-PROFILE

Example

```
esr (config-pbx-profile)# group calling 16
```

Display PBX server information

show pbx peers

This command displays peers connected to PBX.

Syntax

```
show pbx peers
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show pbx peers
```

show pbx registrations

This command displays active registrations on the PBX server.

Syntax

```
show pbx registrations
```

Parameters

The command does not contain parameters.

Required privilege level

1

Command mode

ROOT

Example

```
esr# show pbx registrations
```

34 TECHNICAL SUPPORT

For technical assistance in issues related to operation of Eltex Ltd. equipment, please contact the Service Centre.

Feedback form on the website: https://eltex-co.com/support/

Sevicedesk: https://servicedesk.eltex-co.ru/

Visit Eltex official website to get the relevant technical documentation and software, benefit from our knowledge base, send us online request or consult a Service Centre Specialist in our technical forum:

OOfficial site: https://eltex-co.com/

Technical forum: https://eltex-co.ru/forum

Knowledge base: https://docs.eltex-co.ru/display/EKB/Eltex+Knowledge+Base

Download center: https://eltex-co.com/support/downloads